



[12] 发明专利申请公开说明书

[21] 申请号 97122489.7

[43] 公开日 1998 年 8 月 12 日

[11] 公开号 CN 1190205A

[22] 申请日 97.11.14

[30] 优先权

[32] 96.11.15 [33] US [31] 751,514

[71] 申请人 辛纳普蒂克斯有限公司

地址 美国加利福尼亚州

[72] 发明人 C·N·王 J·C·普拉特
N·P·马蒂克

[74] 专利代理机构 中国专利代理(香港)有限公司

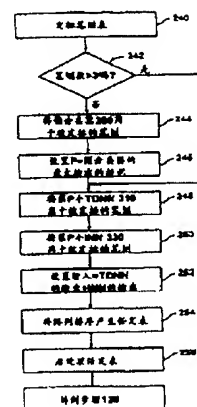
代理人 王 勇 陈景峻

权利要求书 4 页 说明书 35 页 附图页数 14 页

[54] 发明名称 表意字符的递增输入方法

[57] 摘要

递增识别表意字符手写体方法包括：(1) 以自然笔划次序输入至少一个笔划；(2) 将比划提供给递增字符识别器，产生一个假定表；(3) 显示候选字符假定表；(4) 假定表的候选字符中出现正确字符时，对其选择；(5) 当假定表中没有正确字符时，以自然笔划次序至少另外一个笔划；(6) 将该笔划提供给递增字符识别器，产生更新的假定表；(7) 显示更新的候选字符假定表；(8) 更新的假定表的候选字符中出现正确字符时，对其选择；(9) 重复步骤 (5) 至 (8)，直到选出一个正确字符止。



权 利 要 求 书

1. 递增识别表意字符手写体的装置, 包括:

5 划;
一个计算机图形输入板, 用于以自然笔划次序输入表意字符的笔

一个递增字符识别器, 与所述计算机图形输入板相连, 响应部分字符, 产生包含所述笔划的至少一个候选字符的一个假定表;

用于显示所述假定表的装置;

10 用于从所述假定表的所述至少一个候选字符中选择一个正确字符的装置;

当所述假定表的所述至少一个候选字符中没有一个被选择时用于以自然次序给所述递增字符识别器提供至少另外一个单一笔划的装置。

15 2. 根据权利要求1所述的装置, 其特征在于所述递增字符识别器包括:

一个预处理器和一个统计模式分类器。

3. 根据权利要求2所述的装置, 其特征在于所述统计模式分类器包括至少一个标准的神经网络层。

20 4. 根据权利要求2所述的装置, 其特征在于所述统计模式分类器包括一个延时神经网络。

5. 根据权利要求1所述的装置, 其特征在于所述递增字符识别器包括:

多个细分类器, 每个细分类器识别可能字符的一个子集;

一个预分类器, 选择多个细分类器的一个来使用。

25 6. 根据权利要求1所述的装置, 其特征在于所述递增字符识别器进一步包括用于处理所述候选字符假定表的装置。

7. 根据权利要求6所述的装置, 其特征在于处理所述候选字符假定表的所述装置包括用于删除假定的装置。

30 8. 根据权利要求6所述的装置, 其特征在于处理所述候选字符假定表的所述装置包括用于插入假定的装置。

9. 根据权利要求6所述的装置, 其特征在于处理所述候选字符假定表的所述装置包括用于重排假定的装置。

10. 用于递增识别表意字符的装置包括:

用于输入代表表意字符笔划的坐标数据的输入装置;

用于处理所述坐标数据的输入处理器;

响应部分字符并产生具有至少一个候选字符的一个假定表的递增识别器;

显示装置;

所述显示装置上的一个窗口,被划分为显示笔划的第一区域和显示所述至少一个候选字符的第二区域。

11. 根据权利要求10所述的装置,其特征在于所述输入装置是一个在宽度和高度上都小于3英寸的触摸板。

12. 根据权利要求10所述的装置,其特征在于所述输入装置的整个坐标系都被映射到所述窗口的整个坐标系上。

13. 根据权利要求10所述的装置,其特征在于所述窗口进一步包括一个用于显示命令的区域。

14. 根据权利要求13所述的装置,其特征在于所述命令区域位于所述窗口的边沿。

15. 根据权利要求13所述的装置,其特征在于所述用于显示所述至少一个候选字符的所述区域位于所述窗口的边沿。

16. 根据权利要求13所述的装置,其特征在于所述笔划区域被划分为若干子区域,所述输入处理器响应在所述若干子区域的坐标输入。

17. 递增识别表意字符手写体的方法,依次包括以下步骤:

(1) 从一个计算机图形输入板输入表意字符的至少一个笔划;

(2) 提供所述至少一个笔划给包括一个预处理器和一个统计模式分类器的递增字符识别器,所述递增识别器产生包含至少一个候选字符的一个假定表;

(3) 显示包含所述至少一个笔划的所述候选字符假定表;

(4) 当所述假定表的所述候选字符中出现一个正确字符时,从中选择一个正确字符;

(5) 当所述假定表没有一个候选字符是正确字符时,从所述计算机图形输入板输入所述表意字符的至少另外一个笔划;

(6) 将所述至少一个另外的笔划提供给所述递增字符识别器,并产生包括至少一个候选字符的更新的假定表;

(7) 显示包含每个笔划的所述更新的候选字符假定表;

(8) 当所述更新的假定表的所述候选字符中出现一个正确字符时, 从中选择一个正确字符;

5 (9) 重复步骤(5)至(8), 直到从所述更新的假定表中选出一个正确字符为止。

18. 根据权利要求17所述的方法, 其特征在于所述统计模式分类器包括至少一个标准的神经网络层。

19. 根据权利要求17所述的方法, 其特征在于所述统计模式分类器包括一个延时神经网络。

10 20. 根据权利要求17所述的方法, 其特征在于所述递增识别器包括:

多个细分类器, 每个细分类器识别可能字符的一个子集;

一个预分类器, 选择多个细分类器中的一个来使用。

15 21. 根据权利要求17所述的方法, 其特征在于所述产生所述更新的假定表的步骤包括从所述假定表删除至少一个所述候选字符的步骤。

22. 根据权利要求17所述的方法, 其特征在于所述产生一个更新的假定表的步骤包括在所述假定表中插入至少一个另外候选字符的步骤。

20 23. 根据权利要求17所述的方法, 其特征在于所述候选字符有一个次序, 以及所述产生一个更新的假定表的步骤包括在所述更新的假定表上重排至少一个所述候选字符的步骤。

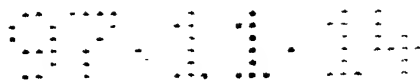
24. 递增识别表意字符手写体的方法, 依次包括以下步骤:

(1) 从一个计算机图形输入板以自然笔划次序输入表意字符的至少一个笔划;

25 (2) 提供所述至少一个笔划给包括一个预处理器和一个统计模式分类器的递增字符识别器, 所述递增识别器产生包含至少一个候选字符的一个假定表;

(3) 显示包含所述至少一个笔划的所述候选字符假定表;

30 (4) 当所述假定表的所述候选字符中出现一个正确字符时, 从中选择一个正确字符;



(5) 当所述假定表没有一个候选字符是正确字符时, 从所述计算机图形输入板以自然笔划次序输入所述表意字符的至少另外一个笔划;

5 (6) 将所述至少一个另外的笔划提供给所述递增字符识别器, 并产生包括至少一个候选字符的更新的假定表;

(7) 显示包含每个笔划的所述更新的候选字符假定表;

(8) 当所述更新的假定表的所述候选字符中出现一个正确字符时, 从其中选择一个正确字符;

10 (9) 重复步骤(5)至(8), 直到从所述更新的假定表选出一个正确字符为止.

说明书

表意字符的递增输入方法

本发明相关于将表意字符输入计算机的一个用户接口。

5 表意字符的计算机输入存在一个难以解决的用户接口问题。例如，在基本的汉字字库中，有5401个字符，每个字符对应一个不同的含义，具有5401个键位的键盘无论从制造还是使用来讲都是不可能的。

10 现有技术中有两种不同的解决方法，第一种采用一个键盘，其中输入一个字符需要一个以上的键位，第二种采用一个图形输入板和一个识别所有字符的软件手写识别系统。

对表意字符有多种键盘输入方法。例如，有一种称为Ziran输入系统的键盘，由加拿大Calgary的Multi-Corp制造，其中每个键位对应一种特定笔划（横、竖、竖折、等等）。为了输入一个字符，用户必须决定哪种形状的笔划最接近他想要的，然后按下相应的键。还有其它键盘输入系统，其中每个键对应一个特定的音或字符的子集（称为部首）。参考例如授予Wong的美国专利5, 131, 766、授予Feng的美国专利4, 669, 901、授予Yong-Min的美国专利4, 684, 926。

20 所有这些键盘输入系统都是不直观的，需要长时间练习才能熟练使用。例如，对Ziran输入系统，初学者在按下任意一个键之前常常要用手指将整个字符写在桌子上，因为哪种笔划实际最接近想要的字符是不直观的，初学者需要加以考虑才能确定。

对亚洲表意字符的输入来说，产生输入困难的另一个原因是字符包含的笔划数。某些字符包括多达30个笔划。

25 有一个能改进Ziran输入系统使用性的调节因子，该系统在每按下一个键后提供多种假定，该假定是多达大约10个字符的一个表，这些字符与目前按下的键一致。通常用户只需要输入一个字符的前四到六个笔划，系统就可以将候选表缩小到一个或两个可能的字符。用户可以在任何时候从表中选出一个候选字符。这种增量方法将输入速度提高了大约2至4倍。

30 输入表意字符的另一种方法是用笔在图形输入板上自然地写出整个字符，例如，参考Crane等人的美国专利4, 561, 105、Greanias等人

的美国专利4, 365, 235和Kitamura的美国专利5, 462, 711. 软件识别系统试图找出与图形输入板上输入的笔划最相配的字符。某些表意字符识别器只识别书写整齐的字符, 然而, 整齐书写整个字符要花费较长的时间, 为了加快字符输入处理, 人们会将笔划连接起来形成草体字。但草体字具有很强的特意性和/或很不整齐, 因此在软件试图识别写的什么时造成很差的准确性。

另外, 现有技术的表意字符识别器的例子或者需要在图形输入板下有一个显示器, 或者需要一个能感应悬在图形输入板上方的笔的贴近的图形输入板, 迄今为止这些技术是使书写笔划与选择指令和/或字符假定有效结合所必须的。然而, 这两种技术都很昂贵, 增加了手写识别产品的费用。

Monroe等人的美国专利4, 829, 583描述了采用图形输入板输入表意字符的笔划的一个系统。用户首先写出字符的第一笔和最后一笔, 系统随后根据被量化9X9坐标网格的这些笔划的起点和终点来识别该字符, 如果根据这两个笔划不能唯一判定字符, 则显示出一列候选字符供用户选择。在由 Monroe等人公开的另一个实施例中, 用户可以输入另外的特定笔划来确定字符。这些另外的笔划可以是第二笔和倒数第二笔、部首的最后一笔或紧随部首的笔划。

采用 Monroe等人公开的系统具有严重的局限性, 主要是因为采用的识别算法过于简单。由于识别算法是一个查询表, 没有公开误差校正手段, 因此, 用户必须严格在系统要求的9X9坐标网格中开始和结束特定的笔划。如果用户在一个相邻的方格中开始或结束笔划, 系统就不能识别该字符。Monroe等人公开的系统在图形输入板上提供一个坐标网格层来帮助书写笔划。然而, 用户常常无法记住所有5401个字符的所有笔划的准确起点或终点。此外, 如果图形输入板很小, 如在触摸板的情况下, 在快速书写时很难正好点在正确的坐标网格方块上。强制要求起始和结束在准确的坐标网格方块上, 既容易出错, 也很不舒服。如果将一个笔划准确开始或结束在正确方块上的成功率是95%, 那么两个笔划之后识别的可能性就只有 $(.95)^4=81\%$, 如果需要六个笔划来识别, 那么采用 Monroe等人的系统的识别可能性就只有 $(.95)^{12}=54\%$ 。如果字符非常复杂, 需要18个笔划来与别的字符区别开, 那么识别的可能性是 $(.95)^{36}=16\%$ 。

显然，Monroe等人的系统必须限制用很少几个笔划来识别字符。本领域的技术人员公知，汉字中第一笔和最后一笔是最关键的，Monroe等人利用了这个事实，因此用户必须首先写出字符的第一笔和最后一笔，随后可能写出靠近部首末尾的关键笔划。

5 然而，被强迫书写字符的第一笔、然后是最后一笔、再然后可能是中间任意某些笔划是很不直观也很容易出错的。如果用户以公知的正确笔划次序开始书写笔划，通常可能需要3到18笔才能确定该字符，因为很多字符具有相同的部首，而该部首可能多达17个笔划。如上所述，如果用户以公知的正确笔划次序书写，Monroe等人的系统就会出现令人
10 无法接受的高出错率，另外，强迫用户以不自然的方式（即与字符自然书写的次序不同）书写字符会包含既不灵活也不适宜的用户接口。

 本发明的一个目的是建立一个用于表意字符的增量输入方法，使用户以公知的正确次序书写笔划，并同时保持高准确率。

 本发明的另一个目的是提供一个与在图形输入板上书写字符一样
15 直观的表意字符输入方法。

 本发明的另一个目的是提供一个比标准的表意字符识别更快也更准确的表意字符输入方法。

 本发明的又另一个目的是提供一个采用低价格图形输入板输入表意字符的方法，该图形输入板不用邻近监测或显示器。

20 本发明的另一个目的是提供一个能够克服现有技术中的某些缺点的增量型表意字符输入法。

 本发明是一个用于将表意字符输入计算机的用户接口方法。该方法包括以下步骤：首先，用户在图形输入板或触摸板上输入表意字符的至少一个笔划，该笔划以表意字符书写的自然次序书写（下称“自然笔划次序”），用这些初始笔划对笔划表进行初始化，并将笔划表提供给一个增量字符识别器，后者产生包括这些初始笔划的候选字符的假定表。
25 然后用户可以通过指示需要的候选字符，从假定表中的候选字符中选出一个正确的字符，如果没有正确的候选字符，用户可以写出字符的另一个笔划，将该另外的笔划加到笔划表中，并将笔划表提供给增量型字符识别器，产生一个新的假定表。该方法继续接收笔划直到用户选择一个假定为止。
30

另外，用户接口可以接收用户的命令，该命令可以用于删除笔划表中的某些或所有笔划，停止识别程序，或发送其它命令给字符识别程序、应用程序或操作系统。

根据本发明的另一个实施例，用户可以在两个不同的窗口交替写入笔划。当在另一个窗口写入一个笔划时，用户指示应该选择写在第一个窗口中的笔划对应的候选字符表中的第一个候选字符。

通过采用触摸板并将选择区域放在窗口的边缘,可以更容易选择命令和候选字符。这种放置容许用户将手指沿触摸板的边缘移动,来加亮各种命令或候选字符。用户不必看他的手指,而是可以通过感觉触摸板的斜面来找出边缘。由于触摸板不能感觉笔或手指的接近,这通常阻碍了触摸板作为绝对方式指示装置的使用性,而边缘放置命令能使触摸板作为一个有效的表意字符输入装置来使用。边缘放置命令允许使用任何没有接近感应或在触摸板下方安装的显示器的输入装置。

15 本发明的递增识别器比现有技术要高级得多，因为它基于统计模式分类器，它们可以用于产生一个更准确的递增识别系统。递增识别器的第一部分是一个预分类器，它将所有可能的候选字符中的候选字符限定为很小的一组，以便加速识别。然后一个细分类器确定写入的部分或全部字符的标识（identity）。用两个神经网络来构成一个细分类器，第一个网络是一个两层的时间延迟神经网络，最适合用于识别部分字符，
20 第二个网络是一个两层的标准神经网络，最适合用于识别完整字符。两个网络的输出被组合起来产生一个对部分字符和完整字符都很适合的系统。然后将两个网络的输出组合排序，产生一个假定表。

25 作为后处理步骤，递增识别器处理假定表，笔划比用户写入的笔划明显少的字符被从假定表中删除，属于假定表上其它字符子集的字符或者被提到表的前面，或者被插在表的前面。

图1是实行本发明的表意字符递增输入法的装置的方框图;

图2是用于本发明的表意字符递增输入法中的一个典型显示窗口的示意图;

30 图3是由本发明的输入处理器执行的初始步骤的流程图;

图4是根据本发明方法启动笔或手指向下处理的一个当前较佳实施例的流程图;

图5的流程图示出了在初始向下点之后笔或手指向下作用期间本发明方法执行的处理步骤;

图6的流程图示出了当用户将笔或手指从触摸板或图形输入板上提起时输入处理器执行的步骤;

5 图7的示意图示出了本发明的显示窗口的另一个实施例, 它用于有经验的用户;

图8的流程图示出了实现图7的用户接口的步骤;

图9的流程图示出了对输入装置是触摸板时较佳的对用户接口的可选增强;

10 图10的流程图示出了图1的增量识别器的一个当前较佳实施方式;

图11是适用于本发明的一个预分类器的结构方框图;

图12a的示意图示出了模糊译码是如何将一个连续标量转换为模糊成员变量中的一个矢量的;

15 图12b的示意图示出了模糊译码是如何将一个0到360度范围内的方向变量转换为模糊成员变量中的一个矢量的;

图13的流程图示出了图11的分类器的方向特征变换 (DFM) 的操作;

图14的方框图示出了适用于本发明的一个时间延迟神经网络 (TDNN) 的结构与布置;

20 图15的方框图示出了适用于本发明的一个基于图像的神经网络 (INN) 的结构与布置;

图16的流程图示出了安排假定表以便增大假定表中的第一个元素作为正确答案的机会;

25 图17的流程图示出了在图16的假定删除算法的较佳实施例中的步骤;

图18的流程图示出了在图16的假定插入算法的较佳实施例中的步骤;

图19的流程图示出了在图16的假定重排算法的较佳实施例中的步骤。

本领域的技术人员会认识到下面对本发明所作的描述是示意性的，不对本发明产生任何限定。本领域的技术人员很容易联想到本发明的其它实施例。

在本发明的说明书和权利要求书中，采用了“自然笔划次序”的用语，在这里表示该语言是其母语的人通常书写一个表意字符的笔划次序。

在本发明的说明书和权利要求书中，采用了单词“笔”，在这里表示由图形输入装置或触摸板感应到的一个位置指定装置。对一个标准的图形输入板来说，它是一个笔尖（stylus），而对一个标准的电容触摸板来说，它是一个手指。

在本发明的说明书和权利要求书中，采用了“统计模式分类器”的用语，在这里表示将输入模式分为若干类中的其中之一的一个装置，这样一种分类器的参数和/或结构由输入模式的统计分析确定。统计模式分类器的例子包括神经网络、径向基函数、分类和回归树、参量贝叶斯分类器、最相邻分类器、局部参数模型、专家混合和多项式分类器。统计模式分类器还可以包括若干采用组合算法对其输出进行组合的统计模式分类器。统计模式分类器的其它例子对模式识别领域的技术人员是显而易见的。

参考图1，其中示出了用于执行本发明的表意字符输入方法的一个装置10。用户采用图形输入板或触摸板14与计算机12进行交互作用。根据本发明的一个当前较佳实施例，用户将采用一个触摸板，因为它的台面面积很小（小于4平方英寸），因此很容易集成到键盘或便携式计算机中。触摸板的输出由执行本发明方法的输入处理器16进行处理。输入处理器与递增识别器18交互作用，在显示器20上绘制图形，并发送字符信息到操作系统或应用程序22。输入处理器通常由运行在计算机12上的软件实现。

参考图2，其中示出了一个输入处理器的典型显示窗口30，窗口30可以是一个标准窗口系统的一个窗口如X窗口或Windows95。窗口30通常是计算机显示器20的一个子集。窗口30的一部分32被保留来显示由增量识别器产生的假定，另一部分34示出用户目前为止写出的笔划，示于窗口30的右侧的再另一部分36保留给按钮，用户由此可以向输入处理器发命令。

根据本发明的较佳实施例，触摸板14的绝对坐标系被转换为窗口30的坐标系，并且窗口30与触摸板具有相同的长宽比，从而用户可以通过接触触摸板上的任何一处来达到整个窗口。示于图2中的区域32、34和36的安排最适合与触摸板一起使用，因为在区域34和36内选择子区域只需要用户将其手指沿触摸板的纵向边沿移动。区域32、34和36的其它安排在本发明范围内也是可能的。

现在参考图3，其中示出了由本发明的输入处理器16执行的初始步骤的流程图。触摸板输入装置14产生数据包，输入处理器16必须处理这些数据包，并对用户的意图作出合适的反应。

在步骤40，输入处理器等待从触摸板14输入的下一个数据包。根据本发明的当前较佳实施例，假定这些数据包被一个单独的执行线程排队，从而输入处理器不需要有多个执行线程。每个数据包具有来自触摸板的X和Y的绝对位置数据，以及一个指示笔（或手指）是否与触摸板接触的信号。

当一个数据包到达输入处理器时，执行步骤42，它以变量penDown存储笔当前是否接触触摸板的信息。然后执行步骤44，其中触摸板上的（X，Y）点被转换为图2的窗口30中的（X'，Y'）点。如上所述，当输入处理器正执行本发明的步骤时，整个触摸板被映射到整个窗口，该窗口与触摸板具有相同的长宽比。在具有相同的长宽比的矩形之间进行转换的方法是现有技术中公知的技术。另一个实施例可以包括标准的图形输入板，其中图形输入板的坐标系被映射到显示器20的整个区域，如现有技术中公知的那样。

步骤44之后，步骤46确定penDown是否为TRUE（真），如果是，输入处理器确定当前数据包被转换到哪个区域（32、34或36），这种确定是现有技术中公知的“命中测试”。区域标记在步骤48被存储在loc变量中。步骤48之后，步骤50审查变量oldPenDown，该变量包括先前数据包的penDown变量值。在程序入口处，变量oldPenDown被初始化为FALSE（假）。如果oldPenDown为TRUE，那么执行步骤52。步骤52将在图9作进一步解释。然后，控制进行到图5，在此执行标准的下笔处理。

如果oldPenDown为FALSE，那么步骤54被执行，通过将长度为3的一个数组变量dc初始化为0，并将变量changeStart初始化为0，在步

步骤54初始化startLoc调节算法。然后控制进行到图4的流程，描述初始的下笔处理。如果在步骤46penDown为FALSE，那么处理器在步骤56检查oldPenDown，如果oldPenDown也为FALSE，则用户不与输入处理器交互作用，系统在步骤58将oldPenDown更新为PenDown，并且控制返回到步骤40。如果在步骤56 oldPenDown为TRUE，那么笔刚刚提起，并如图6所示进行提笔处理。

图4示出了根据本发明的初始下笔处理的一个较佳实施例。笔或手指的初始下落点被转换的区域确定用户接口的行为。如果在区域34下笔，那么用户希望写出另一个笔划，如果笔落在区域32，那么用户是想选择候选字符，如果笔落在区域36，那么用户希望输入命令。本实施例的变化可以由本领域的技术人员联想到。例如，用户可能需要在触摸板上相应于区域34或36的区域作出敲击动作，以便选择候选字符或执行命令。图4执行的较佳实施例仅仅通过在一个区域下笔来指示将要从该区域选择一个对象。

在步骤60，变量startLoc将被装入第一下笔点的标号，它包含在位置变量“loc”之中。变量startLoc将在整个用户接口用作模式标志。如果在步骤62 startLoc为34，那么用户希望写入一个新的笔划，因此步骤64分配一个新笔划，步骤66将当前触摸板的点(X, Y)加到该笔划上。

如果startLoc不是34，步骤68确定是否为32。如果是，那么用户希望选择一个候选字符。然后输入处理器执行步骤70，找出图形显示包含点(X', Y')的候选字符，并高亮显示该候选字符。其它图形接口动作，如按下图形按钮或改变候选字符的颜色也可以用来指示作出的选择。选定的候选字符的标识在步骤72被存储在变量oldCandNum中。如果startLoc既不是34也不是32，那么处理器确定用户希望输入一个命令，在步骤74高亮显示其图形显示包含(X', Y')点的命令，并将选定的命令的标识在步骤76存储在变量oldCmdNum中。

在步骤66、72或76被执行之后，控制返回到图2流程的步骤58，准备处理器接收新的数据包。

现在参考图5，其中的流程图示出了由本发明方法在初始下笔点之后笔落下的时间间隔内执行的步骤。这些步骤将导致笔划被画出或使用户沿触摸板或图形输入板14的左右边沿滑动笔或手指来选择不同的候选字符或命令。

步骤80通过确定startLoc是否是34来确定用户是否在写出笔划，如果是，在步骤82将当前的(X, Y)点加到笔划上，并在步骤84在窗口30上从先前(X', Y')点到当前(X, Y)点画出一条直线。然后控制返回到图2流程图的步骤58，准备处理器接收新的数据包。

5 如果startLoc不是34，步骤86通过确定startLoc是否是32来检查用户是否在选择一个候选字符，如果是，那么步骤88通过检查loc是否是32来确定用户是否仍然按在触摸板的候选区域，如果是，那么步骤90将变量candNum设置为一个候选字符的标识，该候选字符在显示器20的窗口30中的图形显示包含了当前的(X', Y')位置，然后控制进行到
10 步骤94。

如果loc不是32，那么用户没想选择候选字符，因此在步骤92将变量candNum设置为-1，然后步骤94确定oldCandNum是否与candNum相同，步骤94的作用是最大限度地减小在窗口30上的图形操作次数。如果oldCandNum与candNum相同，那么不需要进行图形操作，控制返回到
15 图3的步骤58。

如果oldCandNum与CandNum不同，那么必须改变对候选字符的高亮显示，如果oldCandNum不是-1，步骤96消除对与oldCandNum相应的候选字符的高亮显示。然后，若CandNum不是-1，步骤98高亮显示与candNum相应的候选字符。最后，步骤100用candNum的当前值更新
20 oldCandNum，控制返回到图3的步骤58。

在步骤86，如果startLoc不是32，那么处理器确定用户必定是在选择一个命令。此时执行步骤102至114。步骤102至114与步骤88至100完全类似，只是高亮显示的是命令而不是候选字符，操作的变量是oldCmdNum和cmdNum而不是oldCandNum和candNum。因此，在步
25 骤102，确定startLoc是否是36（指示是否想要一个命令），如果在步骤102确定loc不是36，那么就不是要命令，因此步骤104将变量cmdNum设置为-1，然后控制返回到步骤108。

如果指示了一个想要的命令，步骤106就将变量cmdNum设置为其在窗口30中的图形显示包括当前(X', Y')位置的命令的标识，然后步
30 骤108确定cmdNum是否等于oldCmdNum，如果是，那么就不需要作进一步的图形处理，控制返回到图3的步骤58。

如果oldCmdNum与cmdNum不同，那么必须改变命令的高亮显示。如果oldCmdNum没被设置为-1，步骤110消除对与oldCmdNum相应的候选命令的高亮显示。然后，如果cmdNum没被设置为-1，步骤112高亮显示与cmdNum相应的命令。最后，步骤114用cmdNum的当前值更新oldCmdNum，控制返回图3的步骤58。

参考图6，其中的流程图示出了当用户将笔或手指从触摸板或图形输入板14上提起时图1的输入处理器16采取的步骤。根据本发明的一个当前较佳实施例，提笔是一个信号，表示或者将笔划发送到递增识别器，或者选择一个字符或一个命令。如上所述，另外的实施例可以采取不同的选择行为。例如，在任何时间点击鼠标的左边按键，用户可以指示对表中第一候选字符的选择。

步骤118首先通过检查startLoc是否是34来确定用户是否处于输入笔划模式。如果是，步骤120确定笔划是否太小以至无法处理。如手写识别技术中公知的那样，很小的笔划常常会当做用户不小心画出的笔划而忽略。然而，在识别表意字符时要非常小心，因为有些用户写得非常快，因此他们的笔划被大大缩短。在步骤120的较佳实施例中，确定笔划宽度和高度的最大值。如果这个最大值在触摸板上小于大约1 mil，那么控制返回到图3的步骤58。

如果在步骤120确定笔划并不太小，步骤122将当前的笔划加到包含到目前为止写出的笔划的笔划表中。当处理程序开始时，将该笔划表初始化为空。步骤124将笔划表传送到一个递增识别器18，然后在步骤126从识别器18返回包含候选字符的一个假定表。

递增识别器是一个受到一些限定的识别器。最好是递增识别器在小于250毫秒下操作，以便不以每个笔划之间的长延迟时间干扰用户。另一方面，如果执行一个可选择的中断，那么识别器可以用较长的时间来操作。递增识别器不能采用字符的总笔划数，也不能采用字符最后的大小，原因是这些信息在书写该字符时无法得到。然而，递增识别器可以采用到目前为止写出的笔划或到目前为止写出的字符的大小。

基于字典的识别器（如Crane等人的美国专利4, 561, 105、LeCun等人的美国专利5, 337, 372和Carman II的美国专利5, 454, 046中描述的）可以用作递增识别器，因为字典或参考模式可以扩展到包括表意字符的缩写。另外，递增识别器可以由本领域的普通技术人员采用任意

数目的统计模式分类器，如分类与回归树、神经网络或这些元件的组合来构成。本发明是递增识别器的一个用户接口，不依赖于任何特定的识别技术。

5 作为一种最佳方式，递增识别器可以保存与输入处理器的调用之间的输入笔划表相关的状态。然而，输入处理器必须对笔划表进行存取，以便按照用户的指示增加或删除笔划。

10 作为另一种最佳方式，如果输入处理器16检测到一个下笔数据包的到来，那么在执行识别时输入处理器可以中断递增识别器。下笔数据包指示用户正试图写一个新笔划或选择一个命令或候选字符。在较佳实施例中，递增识别器中断在处理数据包排队（步骤40）的执行线程中执行。排队线程与执行输入处理程序和递增识别程序的线程分开。当数据包从触摸板输入时，排队线程立即对其进行操作。排队线程将当前数据包的上/下状态写入一个存储器存储单元。当执行输入处理程序和递增识别程序的线程正执行递增识别程序步骤时，它周期性地检查这个存储单元。
15 如果该存储单元指示在递增识别程序正被执行时当前数据包是下笔数据包，那么递增识别程序中断，控制立即返回到图3的步骤58。这个中断保证了对窗口30的图形更新不会因递增识别器18的处理而变慢。

中断递增识别程序表示用户可以在某些时候在假定字符表产生之间写出一个以上的笔划。在另外一个实施例中，递增识别程序总可以等待一个以上的笔划才开始产生假定或产生进一步的假定。这个另外的实施例并不是较佳的，原因是用户可以在调用递增识别程序之前完成字符，从而增大了用户接口的复杂性。

25 在步骤126由输入处理器收到的递增识别器的输出是用户正试图画出的表意字符的标识的一个或多个假定。步骤128将在显示器20的窗口30的区域32显示该假定表，以便减小可信度或可能性。由递增识别器产生的假定应该很容易显示在区域32并由用户采用触摸板14选择。在实际应用中，方便显示和选择的要求通常将显示的假定字符限制在少于15个。递增识别器可以产生比区域32能容纳的更多的假定，因此步骤128必须缩短假定表到适当的长度。显示之后，在图3的步骤58系统准备接收新的输入数据包。
30

如果在步骤118 startLoc不是34，那么在步骤130输入处理器确定startLoc是否是36，如果不是，那么可以确定用户正在选择一个字符，

做法是在触摸板或图形输入板14相应于窗口30的区域34的一个区域提笔。然后执行步骤132来确定触摸板上的当前位置是否相应于显示器20的窗口30的选择区域32,如果不是,那么可以确定用户正通过移出选择区域来中断选择程序。因此,控制返回到图3的步骤58。

5 如果触摸板上的当前位置相应于显示器20的窗口30的选择区域32,步骤134发送相应于变量oldCandNum的值的候选字符到操作系统或另一个应用程序。根据在Windows 95下运行的本发明的一个较佳实施例,图1的输入处理器16采用输入方法编辑程序消息与操作系统通信,这在现有技术中是公知的。

10 当在步骤134发送了候选字符之后,步骤136、138和140清除输入处理器16的状态,以便接收新的字符。步骤136使笔划表变空,步骤138停止假定表的显示,步骤140清除显示区域34。然后处理程序控制返回到图3的步骤58。

15 如果在步骤130startLoc是36,那么可以确定用户正在选择一个命令,然后步骤142确定用户是否仍然指示在命令选择区域,如果不是,那么用户正在试图中止命令,控制返回到图3的步骤58。否则,步骤144确定与变量oldCmdNum的值相应的命令。然后步骤146确定是否选定了停止命令,如果是,那么输入处理器在执行了适当的清除后退出。根据当前的较佳实施例,在输入处理器退出之前,它将触摸板置于相对模式,从而用户可以用标准的鼠标仿真方法来操作光标。

20 如果选定的命令不是停止命令,步骤148将根据命令改变笔划表。例如,如果命令是“全部清除”,那么笔划表成为空的。如果命令是“回退”,那么笔划表中的最后一个笔划被删除。

25 步骤150将改变区域34中的显示来反映步骤148中笔划表的改变。然后程序控制返回到图3的步骤58。

本领域的技术人员会认识到,在命令区域也可以加上其它命令。在本发明的系统中“全部清除”和“回退”命令是较佳命令,但也可以加上其它命令。这些命令不必一定要影响笔划表或显示,但可以比如发送命令到操作系统或递增识别器。这种命令的另一个例子是调用短语完成字典,如在现有的商用表意字符输入系统中常常所作的那样。

30 当没有笔划被输入到系统中时,在步骤114就没有假定被显示在区域32或被选择。在一个实施例中,当没有笔划被输入系统中时,各种通

用的汉字、英语字符或标点可以作为假定被显示出来。这种假定的例子可以是空格键、回车、句号或逗号。另一方面，这些假定也可以代表常用的部首。如果用户不写出任何笔划而选择一个常用的部首，这指示他们想启动具有选定部首的字符。在该方面的一个较佳实施例中，存储起来的一个写出的选定部首的例子被插入到笔划表中，并且继续步骤124
5 的处理程序，如同存储的笔划是由用户输入的一样。

现在参考图7，其中示出了本发明图形的另一个实施例。图7的显示窗口是为对本系统有经验的专家设计的。窗口30现在被分为四个区域，窗口30上方的一条是区域32，显示字符假定，有两个区域（34a和34b）
10 用于笔划输入。窗口30底部的一条命令区域36用于选择命令。

如果用户先在一个区域写出笔划，然后不选择候选字符或执行命令而在另一个区域写出笔划，那么输入处理器将自动选择最合适的候选字符并将其送到操作系统或应用程序。这个用户接口能使用户持续地写出字符，不必停下来选择候选字符。尽管较佳实施例是将区域32和36放置
15 在窗口30的边沿，以便于触摸板的使用，但本领域的技术人员很容易想到区域32、34a、34b和36的其它几何安排。

图8的流程图示出了执行图7所示的用户接口的步骤。图8中的步骤替代了图4中的步骤。当采用图7的用户接口时，输入处理器仍然采用图3、图5和图6的步骤。当采用图7的用户接口时，如果startLoc等于34a
20 或34b，那么对图5和图6中startLoc等于34的检查是真的。

步骤160的执行与图4的步骤60完全一样，当步骤160完成之后，步骤162、164、166和168计算下面的逻辑条件：（startLoc=34a 并且先前的startLoc=34b）或（startLoc=34b 并且先前的startLoc=34a）为真
25 吗？如果是，那么用户已经从显示器20的窗口30中的区域34a移动了34b（或者正好相反）并希望以缺省方式选择最好的候选字符。在这种情况下，步骤170发送最好的候选字符给应用程序或操作系统，并且步骤172、174和176以图6处理程序的步骤136、138和140的相同方式执行必要的清除步骤。

如果逻辑条件为假并且startLoc=34a或startLoc=34b，或者当步骤
30 176被执行之后，步骤178和180被执行。这些步骤的执行与对图4的步骤64和66所描述的一样，它们以正常的方式开始一个新的笔划。

如果在步骤164startLoc既不是34a也不是34b，那么步骤182确定startLoc是否是32，以及由此判定是否选择了一个候选字符或命令。如果startLoc是32，就执行步骤184和186。这些步骤与图4的步骤70和72一样，是以正常方式执行的。当图8的步骤184和186结束之后，程序控制返回到图3的步骤58。

如果在步骤182确定startLoc不是32，那么执行步骤188和190。这些步骤与图4的步骤74和76相同，并且是以正常的方式执行的，当图8的步骤188和190结束之后，程序控制返回到图3的步骤58。

图9的流程图示出了当输入装置是触摸板时对较佳的用户接口的增强。由于触摸板没有接近感应，也没有集成化的显示器，用户可以将其手指放在与要求的区域32、34和36都不对应的不正确位置，这会给例如希望选择一个候选字符而不是开始一个新的笔划的用户带来麻烦。图9的步骤将试图通过测定在所有区域花费的时间量，来以启发的方式确定用户想要选择哪个区域。如果在下笔的时间间隔用户在指示一个区域时花费的时间比其它所有区域都多，那么可以肯定用户是想指示该区域。

图9的步骤在输入处理器执行图3中的步骤32时执行。步骤198首先通过检查changeStart变量来确定startLoc是否已经作了调节。为了避免使用户产生误会，对每个笔划startLoc最多只修改一次。如果changeStart不是零，那么算法已经改变了startLoc，控制返回到图3的步骤58。否则，步骤200确定当前的(X', Y')点是否在窗口30的区域34。如果这样，那么测量在区域34花费的时间的计数器dc[0]在步骤202被递增。步骤204确定初始下笔点是否在区域34之外，并确定在区域34花费的时间量(dc[0])是否超出在区域32花费的时间量(dc[1])和在区域36花费的时间量(dc[2])之和。如果这两个条件都是真的，那么用户想要画出一个笔划，而不是选择一个命令或候选字符，因此，startLoc在步骤206被复位为34，并在步骤208开始一个新的笔划。如果其中一个条件为假，那么控制返回到图5的步骤80。

如果步骤200确定loc不是34，那么步骤210确定当前点是否落在区域32。如果这样，那么类似于步骤202至206执行步骤212至216，但这时处理区域32和dc[1]而不是区域34和dc[0]。步骤216之后，步骤218复位当前笔划：如果有一个当前笔划，笔划的内容被删除并且不再显示该笔

划。然后步骤220将变量oldCandNum设置为-1，强迫在图5中高亮显示新的候选字符。

如果在步骤210 loc被确定不是32，那么类似于步骤202至206执行步骤222至226，但这时处理区域36和dc[2]而不是区域34和dc[0]。步骤226之后，步骤228复位当前笔划并且步骤230将变量oldCmdNum设置为-1，强迫在图5中高亮显示新的命令。

当步骤208、220或230被执行之后，在步骤232消除所有当前的高亮显示，并且步骤234将变量changeStart设置为1，指示startLoc已经被改变。然后控制返回到图5的步骤80。

图10的流程图示出了实现图1的递增识别器的较佳实施例。递增识别器18的功能是接收构成部分或完整字符的笔划表并产生该字符的标识的假定表。根据本发明的当前较佳实施例，递增识别器可以从5401个最常用的传统汉字产生假定表。本领域的技术人员会注意到本发明的递增识别器很容易被扩展到包括传统汉字集合的更多字符，或扩展到不同的字符集合，如简体汉字集合、日文汉字或朝文字符集合。

在数千个字符标识之间进行选择，即使对现代化个人计算机而言，也是一个计算密集的任务。由于在本发明的较佳实施例中，要求在每个写出的笔划之间确定递增识别器的值，因此递增识别器必须很快，从而，本发明的较佳实施例分两个阶段识别部分字符：首先将字符集分成可能相互重叠的许多小组，根据本发明的一个当前较佳实施例，5401个字符被分为32个重叠的小组，一个预分类器260确定该部分字符最可能属于哪个小组，然后一个细分类器从小组成员中产生包括最可能的字符的一个假定表。

细分类器本身有两部分：一个最适合用于识别部分字符的神经网络和一个最适合用于识别完整字符的神经网络。已经发现有些用户会在字符结束之前停止，而另一些用户会一直继续到底。通过采用这两个神经网络的输出，可以满足两类用户。相信采用神经网络来识别部分字符，以及两类神经网络的组合，是本发明中具有新颖性的技术。

现在参考图10，其中的流程图为图1的递增识别器的操作提供了详细的示意。首先，步骤240对笔划表定标，以使得字符位于(0,1) x (0,1)的方框中。在较佳实施例中，步骤240保持笔划的长宽比，以便不弄歪该部分字符。更详细地说，笔划表在长宽方向上等同地定标，

并被转换, 这个步骤的结果是被定标的笔划沿X和Y的方向之一从0延伸到1, 并且中心位于沿另一轴的(0, 1)间隔内. 本领域的技术人员显然知道如何建立一个软件系统来实现该定标.

步骤242通过确定笔划数是否大于三来确定是否应该调用预分类器. 在较佳实施例中, 预分类器只检查前三个笔划, 因为前三个笔划几乎总有足够的信息将字符假定限定为单一的一组. 本领域的技术人员显然知道, 可以给预分类器提供更多的笔划, 但这会增大计算费用. 如果步骤242确定有三个或少于三个笔划, 步骤244将预分类器260用于被定标的笔划.

预分类器的结构示于图11, 其操作将结合该图来进行描述. 在较佳实施例中, 预分类器的输出是长度为32的一个向量, 其中的每个元素反应该部分字符作为相应小组的一个成员的置信度. 步骤246随后确定预分类器的最大输出的标识并将其存储为变量P. 在存储器或计算机的磁盘上存在一个32个细分类器的阵列. 预分类器确定第P个细分类器应该被用于产生假定表.

当用于识别所有字符时, 采用预分类器来限制细分类器的检索是公知的, 例如, 可以参考《IEEE 模式研究与机器智能学报》1993年2月第15卷第2期第180至185页, Yih-Tay和Wen-Hsiang Tsai的论文“通过分裂与组合用于联机汉字识别的属性字符串匹配”(“Attributed String Matching by Split-and-Merge for On-Line Chinese Character Recognition”). 然而, 采用预分类器来限制对一个部分字符的标识的检索被认为是具有新颖性的.

此外, 现有技术中公知可以采用两个以上的分类步骤来识别表意字符, 换句话说, 每个细分类器本身可以包括一个预分类器和若干细分类器. 采用两级以上分类的现有技术的例子可以参考《模式识别》1991年第24卷第8期793-799页, T.Kumamoto、K.Toraichi、T.Horiuchi、K.Yamamoto和H.Yamada的论文“手写汉字字符识别中备选字符的加速选择”(“On Speeding Candidate Selection in Handprinted Chinese Character Recognition”). 在极端情况下, 整个递增识别器可以由一个分类器树构成, 其中的每个分类器又进一步限定可能的候选字符表.

在较佳实施例中, 每个细分类器包括两个神经网络: 一个时间延迟神经网络(TDNN) 310和一个基于图像的神经网络(INN) 320. TDNN

310最适用于识别部分字符。TDNN是现有技术中公知的，在1989年3月《IEEE声学、声音和信号处理学报》中，Waibel、Hanazawa、Hinton、Shiano和Lang的论文“使用时间延迟神经网络的音素识别”（“Phoneme Recognition using Time-Delay Neural Networks”）对此做了描述。

5 TDNN的结构将在后面作描述。

再参考图10，步骤248将第P个TDNN用于被定标的笔划，来得到一个输出向量，它包含说明笔划表中的字符标识是某一个值的置信度。这个输出向量的长度随P而变化。在当前较佳实施例中，这些长度的范围是从22到538。采用TDNN识别部分表意字符被认为在本发明中具有新
10 颖性。

INN 320最适用于识别完整的字符。INN是一个标准的多层神经网络，它采用了基于图像的特征。INN的结构示于图13，并将参考该图进行描述。步骤250将第P个INN用于被定标的笔划，来得到一个置信度的输出向量，它与从TDNN得到的输出向量类似。INN输出向量与TDNN
15 输出向量具有相同的长度。步骤248和250可以以任意次序执行。

然后步骤252将TDNN和INN的输出向量相加。相加是组合神经网络结果的最简单方法，但本领域的技术人员知道可以采用更复杂的技术，如《神经计算机》1991年第3卷79-87页中Jacobs、Jordan、Nowlan和Hinton的论文“本地专家的适应性混合”（“Adaptive Mixtures of Local
20 Experts”）所描述的技术。

步骤254随后以递减的次序排序相加的输出向量。相加的输出向量中的位置相应于字符的标识。因此，找出前N个值在相加的输出向量中的位置等于产生了一个长度为N的假定表，在当前较佳实施例中，N等
于10。

25 然后步骤256根据各种规则操作假定表，以便增大假定表中的第一个元素是正确答案的机会。这一后处理过程包括三个步骤，将分别在图16、图17或图18中一一进行描述。随后程序控制转移到图6的步骤128。

预分类器的当前较佳实施例的结构示于图11，该预分类器260是一个TDNN，其中TDNN的第一层262对多个输入重复，如现有技术中公知的那样。在TDNN的现有技术应用中，TDNN第一层262的每个备份相应于语音的一个时间片（参考Waibel等人的论文），或一小段手写体（参考Guyon等人的美国专利5, 105468）。在本发明中，第一层262的每个
30

备份相应于一个表意字符的一个完整笔划。第一层262的每个备份被连接到与一个输入笔划相关的预处理器。采用其中第一层的每个备份相应于一个完整笔划的TDNN被认为是具有新颖性的发明。

更详细地说，笔划264-1至264-3的每一个被分别提供给TDNN的预处理器，有两个预处理模块，其输出被提供给第一层262的每个备份。第一模块是一个几何特征（GF）模块266，该GF模块266计算先前笔划的首末点和当前笔划的首末点之间的几何关系。例如，最高的GF模块266计算笔划1（264-1）和笔划2（264-2）之间的关系。GF模块的特定输出被提供给与笔划2相应的第一层262的备份。下面对由GF模块266执行的计算进行描述。

GF模块266在计算中采用下面四个位置：

PB=先前笔划的第一点

PE=先前笔划的最后一点

CB=当前笔划的第一点

CE=当前笔划的最后一点

然后GF模块266计算下面四个向量：

$D1 = PB - CB$ ； $D2 = PB - CE$ ； $D3 = PE - CB$ ； $D4 = PE - CE$

四个向量的每个元素随后被模糊编码，输入变量的模糊编码使神经网络的训练变得更加容易。一个模糊编码将某些范围的原始连续标量变量转换为如图12a所示的变量的一个向量，这个向量的每个成分代表变量的一个子区域。每个向量成分的值是对原始变量和子区域中心之间的匹配的测量。采用模糊成员函数对连续变量进行编码是公知的现有技术，例如在Prentice-Hall公司出版的（1991）Bart Kosko著的“神经网络和模糊系统”（“Neural Networks and Fuzzy Systems”）一书中第27页作了描述。

作为图12a所示的模糊编码的一个例子，如果一个变量要用编码区域[0, 1]的5个成员函数进行编码，那么5个变量代表与点{0.0, 0.25, 0.5, 0.75, 1.0}的接近程度。

原始值0.1（图12a的A点）随后产生0.6, 0.4, 0, 0和0的模糊编码。原始值0.4（图12a的B点）随后产生0, 0.2, 0.8, 0, 0的模糊编码。与此类似，原始值0.75（图12a的C点）产生0, 0, 0, 1, 0的模糊编码。

位于模糊编码区域之外的原始值与该区域最接近末端的值一样编码，参考图12a的“D”和“E”点可以看出，值-0.2（图12a的D点）变为1, 0, 0, 0, 0, 而值1.4（图12a的E点）变为0, 0, 0, 0, 1。

5 由上面的描述和参考现有技术，本领域的技术人员显然可以了解如何
用软件实现模糊编码。

对GF模块266，二维向量变量D1至D4的每个成分由均匀分布在从-1到1的间隔的八个模糊成员函数进行编码，换句话说，成员函数集中在{-1.0, -0.71, -0.43, -0.14, 0.14, 0.43, 0.71, 1.0}，从而由GF模块产生了总共64个输出，其它方案和编码区域也是可行的。

10 第二预处理模块是一个方向特征映象（DFM）268，DFM利用笔划产生一个三维张量，该张量包括笔划在何处保持了一个一定的方向的空间信息。1988年在罗马召开的第九届模式识别国际会议记录汇编《IEEE计算机协会学报》第168至171页Jun Tsukumo和Haruhiko Tanaka的论文“使用非线性标准化和相关方法对手写汉字字符分类”

15 （“Classification of Handprinted Chinese Characters Using Non-linear Normalization and Correlation Methods”）对DFM作了描述，LeCun等人的美国专利5, 337, 372图2至图6也对DFM作了进一步的描述。在本发明中，在DFM中执行的处理步骤示于图13。

20 在图11中，每个第一层262是一个标准的神经网络层，在较佳实施例中
有80个输出神经。第二层270是一个标准的神经网络层，对总共240个输入，它接收第一层262的所有三个备份的输入。第二层270有32个输出神经，每个对应一组特定的字符标识和一个特定的TDNN和INN。

25 为了本发明公开的目的，一个标准的神经网络层执行将一个加权矩阵乘以一个输入向量的操作，并将产生的向量施加一个S形非线性处理，得到一个输出向量。在Addison-Wesley公司出版的（1991）由J.Hertz、A.Krogh和R.Palmer所著的标准教科书“神经计算理论导论”（“An Introduction to the Theory of Neural Computation”）第6章中对标准的神经网络层作了全面的描述。

30 图13的流程图示出了在用于一个笔划的本发明的DFM中执行的步骤。图13中的步骤280对笔划再取样，来提供在笔划点之间具有规则间隔的一个笔划。当笔划被再取样时，采用了在现有的数据点之间的线性插值，这样一种再采样步骤是现有技术中公知的，例如Guyon等人的美

国专利5, 105, 468就给出了这样的例子。线性插值是众所周知的, 在许多研究教科书中都作了描述, 例如可以参考剑桥大学出版社1992年第二版第114页Press, Vetterling, Teukolsky和Flannery的“C语言中的数值方法”(《Numerical Recipes in C》)。在当前的较佳实施例中, 每个笔划样本值之间的距离是0.06。

再采样之后, 步骤282将当前点pt初始化为再采样笔划中的第一点。步骤284随后将一个三维张量变量A_ijk初始化为0。DFM的主循环从步骤286开始, 这里x变量被装入当前点pt的x位置, 在步骤288将y变量装入当前点pt的y位置。然后步骤290确定该笔划在当前点pt的方向 θ 。确定当前点的方向是现有技术中公知的, 在Guyon的美国专利5, 105, 468的图7中作了描述。

然后步骤292产生x位置、y位置和 θ 方向的模糊编码, 并将该编码分别放在向量变量X_i、Y_j和 θ_k 。根据当前的较佳实施例, 位置x和y被编码为在从0到1的间隔均匀分布的5个成员函数, 方向 θ 被编码为相应于八个主要方向, 即0、45、90、135、180、225、270和315度的八个成员函数, 这八个成员函数在图12b中以F_0、F_45、F_90、F_135、F_180、F_225、F_270和F_315示出。处于0度(F_0)的的成员函数是环绕式的, 因此它与315度(F_315)的成员函数相邻, 如图12b所示。成员函数的环绕使 θ 的模糊编码成为一个没有间断的平滑函数。

然后步骤294用X_i、Y_j和 θ_k 的外积构成一个三维张量, 然后将该张量加到变量A_ijk上。步骤296随后通过确定当前点pt是否是再采样笔划的最后点来结束循环, 如果是, 那么步骤298将A_ijk作为DFM的结果, 否则步骤300将当前点pt设置为笔划的下一点。

在较佳实施例中, DFM 268的三维张量中的所有200个输出值都被当做一个有200个元素的向量, 并提供给第一层262的一个备份。

显然, 本领域的技术人员知道, 在不脱离本发明精神的前提下, 可以改变DFM的参数, 这种参数可以包括用于产生DFM的角分辨率、空间分辨率和确切的模糊成员函数。

如果不到三个笔划被提供给预分类器260, 那么仍然希望一个灵敏的输出, 因此, 较佳实施例用信号表示缺少一个给第一层的输入, 做法是不调用具有一个或多个失去的输入笔划的任何GF或DFM, 相反, 将一个具有正确长度的向量填上0并提供给输入层262的相应备份。对相应

于第一笔划的第一层262的备份来说，从没有一个先前的笔划被提供给GF模块266，因此，GF模块从不被调用，填上0的一个向量总是被提供给第一层262的第一备份。

5 预分类器 260的第一层262和第二层270采用J.Hertz、A.Krogh和R.Palmer的标准课本第六章中描述的标准的向后传播算法来训练。如现有技术中训练TDNN所公知的那样，只有一组存储的第一层加权，它由第一层的所有三个备份提供的梯度（gradrent）进行更新。对TDNN的训练在Waibel等人的论文中作了描述。

10 预分类器260的训练集包括递增识别器应该识别的每个字符的头三个笔划的许多样值。采用标准的联机向后传播，每个样值被提供给预分类器的输入，一个需要的输出向量被提供给预分类器的输出，并且第一层262和第二层270的权被更新。与输入例子所属的组相应的输出被训练为高，所有其它输出被训练为低。如果字符的总笔划小于三，整个字符的一个采样被加到该训练集。

15 训练预分类器260需要将字符指派给要求的小组。将5401个字符指派给较佳实施例的32个预分类器组示于下面的附录A。每个组有一个标记，并且示出了属于该组的字符和该组中的字符数。附录A中的分配不是一个自动处理过程，具有类似的前三个笔划的字符被手工分配到同一小组。对一个不同的字符集或语言，这一手工分组程序必须在可以训练预分类器之前再次执行。这一手工分组程序对本领域的技术人员而言是显而易见的。

20 采用具有这些特征集的一个两层TDNN是执行预分类的一个结构，这一点对本领域的技术人员是显而易见的。TDNN的其它变化，如具有两层以上的一个网络或具有不同特征集的TDNN也可以构成有效的预分类器。总而言之，预分类器可以由本领域的技术人员用任何标准的统计模式分类器构成，如分类和回归树、神经网络、径向基函数（radial basis function）或专家组合。预分类器也可以采用笔划匹配算法来构成。本发明不依赖于特定的识别技术，预分类器本身可以由一个以上的模块组成，包括多个特征检测和多个识别器子模块。预分类器也可以在递增识别器次序调用之间保持状态，以便使总的计算量最小。

30 图14的方框图示出了在本发明中作为细分类器的TDNN 310的结构。除了TDNN 310接收三个以上的笔划外，TDNN 310的结构非常相似

于预分类器260。根据本发明的一个当前较佳实施例，TDNN 310接受高达25个笔划。DFM268和GF模块266与图11的预分类器260中的相同。与在预分类器情况下相同，当没有足够的笔划提供给一个特定的DFM或GF时，那个特定的预分类器的输出是零。在较佳实施例中，TDNN 310
5 包括第一层312的25个备份，该第一层是一个标准的神经网络层。第一层312与预分类器的第一层262具有不同的权。每个第一层312具有20个输出神经。第二层314是一个标准的神经网络层，它接收来自第一层的所有25个备份的输入，总计有500个输入。第P个TDNN310的第二层中的输出数等于第P个细分类器组中的字符标识数。32个不同的TDNN的
10 每一种情况具有不同的第一和第二层权值。

对TDNN310的训练与对预分类器260的训练有些不同。根据较佳实施例，对细分类器组中的每个字符，训练组包括笔划子集，其笔划数从三到字符总笔划数。这些训练子集从第一笔开始，并包括按训练对象书写的次序排列的笔划。如果一个字符具有不到三个笔划，那么整个字符
15 的一个备份被加到训练集。另外，每个整体字符的一个样值被两次额外加到训练集，以便提高正确识别整体字符的可能性。

例如，如果一个字符采样具有5个笔划，那么5个笔划子集被加到该训练集：第一至三笔、第一至四笔和第一至五笔，重复三次。每个笔划子集被标记它所属的字符的标识，相应于该标识的输出被训练为高，所有其它输出被训练为低。因此，TDNN310被训练为识别在细分类器组中具有三个或三个以上字符笔划的所有可能的自然次序子集。
20

一旦预分类器组的指派完成并且预分类器260的训练结束，将字符指派给细分类器组是一个自动过程，整体字符集的训练集中的每个采样的前三个笔划被产生，同样，如果在字符采样中不足三个笔划，就采用
25 整个字符。根据产生的特定字符集中的每个采样评估预分类器260，并记录最大的输出标识M。对每个字符标识，如果值M出现次数多于一个阈值，那么该字符标识被指派给第M个细分类器组。在较佳实施例中，对包含每个字符的40个采样的一个训练集，该阈值是一，这样，一个字符标识可以指派给一个以上的细分类器。字符标识被指派的组是当具有
30 该标识的字符出现时预分类器会选择的组，这样，通过用一个以上的细分类器识别字符，预分类器误差可以得到补偿。细分类器指派阈值不是零，从而可以忽视在训练集中由于书写潦草所带来的影响。

一旦字符的标识在第M个细分类器组中被确定，就可以采用上面描述的对第M个细分类器组的标准的向后传播和整体训练集来训练第M个TDNN310。

正如在预分类器情况下那样，采用具有DFM和GF特征的一个两层TDNN是执行细分类的一个结构，TDNN的其它变化，如具有两层以上
5 的一个网络或具有不同特征集的一个TDNN也可以构成有效的细分类器。总而言之，细分类器可以由本领域的技术人员用任何数量的统计模式分类器构成，如分类和回归树、神经网络、径向基函数或专家组合。细分类器也可以采用笔划匹配算法来构成。本发明不依赖于特定的识别
10 技术。TDNN也可以在递增识别器次序调用之间保持状态，以便使总的计算量最小。TDNN的其它速度最优化，如采用稀疏矩阵向量乘法，对本领域的技术人员是显而易见的。

图15示出了INN 320的结构，它最适合识别完整的字符。INN也帮助识别不按自然笔划书写的字符和草写体字符。在定标的笔划表中的所有笔划被提供给INN。有两个为INN计算的特征集，首先，计算
15 DFM322，该DFM322与图11的DFM268具有不同的参数，DFM322产生一个5x8x8张量，采用五个成员函数对方向编码：F_45、F_0、F_315、F_270和F_225，其方向在这些成员函数之外的很少几个笔划点被忽略，以减小INN320的大小。8x8分辨率仍然覆盖[0-1]x[0-1]矩形。因为
20 提供了所有的笔划，而不只是一个笔划，因此增大DFM322的空间分辨率是必须的。所有笔划DFM322的输出只是图13中用于每个单独的输入笔划的步骤的输出的和。DFM输出中的所有320个象素被作为一个单一向量提供给标准的神经网络324。

EPM预处理模块326提供关于笔划末点位置的信息给神经网络
25 324。对每个笔划的第一点和最后一点，EPM326分别产生这些点的X和Y位置的模糊编码：X_i、Y_i。根据当前的较佳实施例，这些模糊编码的每一个由均匀分布在从0.0至1.0的间隔的五个成员函数产生。下一步，EPM326用X模糊编码和Y模糊编码的外积为每个第一点和最后一点构成一个图像B_{ij}：

30
$$B_{ij} = X_i * Y_j$$

然后，EPM326将所有第一点和最后一点的 B_{ij} 相加，并将相加的值传送到神经网络324。在较佳实施例中，有25个被当做一个向量处理的输出值。

5 根据本发明的一个当前较佳实施例，标准神经网络324包括两个标准的神经网络层，具有375个输入、200个隐蔽单元和与相应的TDNN310具有相同数目的输出。该INN320由标准的向后传播训练。在第M个INN的训练集中的采样与在第M个TDNN的训练集中的采样相同。对包含一个字符的所有笔划的笔划子集，响应于该字符标识的输出被训练为高，而所有其它输出被训练为低。对包含部分字符的所有其它笔划子集，所有输出被训练为断开。因此，INN被训练为只响应完整的字符，而当字符不完整时不加上一个有效的输出信号。

与TDNN类似，32个不同的INN的每种情况在神经网络324中具有不同的权。标准的神经网络324可以以现有技术中公知的方式采用稀疏矩阵向量乘法来达到最优化。

15 图10的步骤256是对假定表进行后处理的一个步骤。在较佳实施例中，有三个构成步骤256的后处理步骤，如图16所示，当然本领域的技术人员还可以想到其它的可能性。这些后处理步骤包括步骤330的假定删除算法、步骤332的假定插入算法和步骤334的假定重排算法。

20 图17的流程图示出了假定删除算法330的较佳实施例中的步骤。假定删除算法的作用是通过与到目前为止写出的部分字符的笔划数相比删除笔划数太少的假定，例如，如果用户写了8个笔划，那么递增识别器可以很安全地删去肯定只有7个或少于7个笔划的假定。假定删除算法需要一个数据阵列，它是假定可能具有的最大笔划数，这个阵列是凭经验从训练表中推导出来的，通过对一个特定字符标识的所有采样找出最大笔划数，来推导出阵列中的每个元素。显然，对本领域的技术人员来说，假定的删除也可以采用其它标准。

25 在图17中，图16的步骤330从步骤340开始，这时将计数器变量i设定为假定数N，在较佳实施例中是20，然后控制进行到步骤342，检查计数器变量是否仍然大于0，如果不是，那么所有假定都已经处理完成，控制可以进行到图16步骤332的假定插入算法，后者将在图18作详细的描述。

步骤344确定M, 即第i个假定的标识最大笔划数, 然后步骤346检查变量M是否小于变量numStrokes, 后者是用户在当前部分字符中写出的总笔划数。如果numStrokes大于M, 那么步骤348从假定表删去第i个假定。在任何情况下, 控制都进行到步骤350, 对计数器变量i递减, 以便考虑下一个假定。

对本领域的技术人员来说, 显然有许多用软件实现假定删除算法的不同方式。图17的算法是示例性的, 本领域的技术人员知道本发明并不局限于以上面公开的方式实现假定删除算法的实施例。

图16的假定插入算法332详细示于图18的流程图。假定插入算法被用来插入是假定表中其它假定的严格子集的假定, 许多表意字符是彼此的子集, 也就是说, 用户可以通过写出一定数量的笔划来构成一个字符, 然后通过加上更多的笔划来构成一个不同的字符。如果用户写出的一个字符是另一个字符的子集, 那么递增识别器可以产生包含更多复杂字符而不是简单的子集字符的一个假定表, 产生这种情况的原因是因为当子集字符被呈现给递增识别器时, 许多复杂字符与简单的子集字符具有类似的置信度。当假定表的长度不足以包含子集字符时, 该子集字符被舍去。

重要的是, 因为用户没有其它方式输入子集字符, 因此当子集字符的所有笔划被输入系统时, 递增识别器突出地显示该子集字符。因此, 假定插入算法在检测到包含该子集的许多假定时, 试图将该子集字符作为第一假定插入。

假定插入算法需要一个以字符标识作为索引的表, 表中的每个条目是作为被索引的字符的子集的一列字符。

图18的详细的假定插入算法从步骤360开始, 其中变量s包括从子集表中取出的第一(即最好的)假定的子集表列, 步骤362确定该表列的长度并将其存储到计数器变量i, 然后步骤364确定计数器变量是否大于0, 如果不是, 那么假定插入算法结束, 控制可以进行到假定重排算法(图16的步骤334)。

如果计数值大于0, 那么在步骤366审查表列中的第i个条目, 确定s[i]是否是前5个假定的至少3个的一个子集, 这个规则是根据经验确定的, 本领域的技术人员也可以建立其他类似的规则。如果s[i]是一个通

用的子集，那么控制进行到步骤368，这时变量M被设置为字符s[i]的最大笔划数。

5 步骤370确定作为部分字符输入的当前笔划数是否大于M，如果不是，那么在步骤372将s[i]加到假定表的第一个位置，如果作为部分字符输入的当前笔划数大于M，或者步骤366的判定结果是s[i]不是前5个假定的至少3个的一个子集，那么该子集不应该被显示出来，并在步骤374递减变量i，然后程序控制返回步骤364，再确定是否i>0。

10 对本领域的技术人员来说，显然有许多用软件实现假定插入算法的不同方式。图18的算法是示例性的，本领域的技术人员知道本发明并不局限于以上面公开的方式实现假定插入算法的实施例。

假定重排算法（图16的步骤334）被详细地示于图19的流程图中。该假定重排算法的作用与假定插入算法类似，即如果在假定表中找到了一个子集字符，就将其提前。然而，由于对假定表重新排序的步骤不如插入一个假定那么严格，因此与插入一个假定相比，重排算法在将一个
15 假定提前时要求的根据较少。

图16的假定重排算法从图19的步骤380开始，这时将计数器变量i设置为1，该计数器变量i将引用当前正被考虑的假定。在较佳实施例中，前10个假定可能会被重新排序，当然也可以考虑其它数目的假定重排。

20 然后步骤382确定计数器变量i是否小于或等于10，如果不是，那么假定重排算法完成，控制将返回到图6的步骤128。否则，控制将进行到步骤384，这时将变量s设置为第i个假定的子集列表。步骤386将计数器变量j设置为变量s的列表长度。计数器变量j将对列表s进行重复。步骤388检查计数器变量j是否大于零，如果不是，那么对第i个假定的检查结果，控制进行到步骤390，这时计数器变量i递增，然后控制返回步骤
25 382。

如果计数器变量j大于零，那么步骤392检查列表s的第j个元素（s[j]）是否出现在假定表中，如果没有，那么它不能重排，控制进行到步骤394，这时j被设置为j-1。否则，步骤396将变量p设置为假定表上的位置s[j]。然后步骤398通过确定变量p是否大于变量i来检查子集字符
30 s[j]是否比目前检查的假定具有较低的置信度。

步骤398也确定子集字符s[j]的置信值是否高到足以使其出现在前十个假定中（步骤398中的数字10可以改变，这对本领域的技术人员是显而易见的）。如果步骤398的两个逻辑比较都是假的，那么子集字符s[j]不应该被提前，控制进行到步骤394。否则，步骤400从假定表删去s[j]，而步骤402将s[j]插入假定表的位置i，该位置正好在当前假定的上面。步骤402保证一个可能的子集字符出现在包含该子集字符的一个更复杂的字符前面。

步骤404将计数器变量i递增1，以对一个新的假定被插在表中当前假定之前的情况进行补偿，然后控制进行到步骤394，这时计数器变量j被递减，以便检查假定i的下一个子集。

对本领域的技术人员来说，显然有许多用软件实现假定重排算法的不同方式。图19的算法是示例性的，本领域的技术人员知道本发明并不局限于以上面公开的方式实现假定重排算法的实施例。

对本领域的技术人员来说，显然可以采用其它标准来重排假定表，例如，如果两个或多个假定具有类似的置信级别，它们可以根据笔划数或字符出现的频率来重排。

本领域的技术人员要认识到，尽管上面对本发明的实施例和应用作了示意和描述，但除此之外的许多改进，只要不脱离本发明的概念，都是可行的，因此，除了权利要求书之外，本发明不受任何限定。

字符分组表(预分类)

竹部:

製怎箭笙筠箬箛版劇獎
失靠筒筏掣笠倚牌卡裝
算符氧筍鉢甥牝片處將
告朱性笛筵笞簍蘭上顧
知智籬籬篋篋高篠弁彪
節築牠迭羅箋篋弱迨虔
造笑箴簾篆笄筵草箴虞
制牛卸乞氣憤枯篋奮遠
午簽牧氣壯篋氣笄怠虐
物筆筋篤箬篋氣笄矣叔
先範篇矩簾籬譚篋半虞
氣簡龍竿篋籬篋篋篋篋
每答氛矢黃箴篋缶九虛樂
管缺迄缸篋篋篋篋篋篋
第舞籃飾篋篋竿篋參膚簪
特籌贊矯低篋篋篋篋篋
無籍籬笨篋篋篋篋篋篋
等短繁矮乍竺簪簪簪督壯
生策箱簿箔篋篋篋篋篋
年竹致篋篋篋篋篋篋篋

皿部:

般艦粵舵免魷鰓邱丘
航鬼魄扉解酥魷兵彙
追艘皂舫刺鱗鰓印白鮑
顧鳥駒蝮蝮蟹蟹鰓鰓鰓
永皇扇帛魷鰓鰓兒邱皮
良郎魷魷魷魷魷魷魷
息遍魷魷魷魷魷魷魷
船泉魁版魷魷魷魷魷
房輿帥舫舫舫舫舫舫舫
白迫般魷魷魷魷魷魷
盤朗擎舵魷魷魷魷魷
戶島扁邊榮逸魷魷魷魷
樂歸舟舫舫舫舫舫舫舫
師啓舫舫舫舫舫舫舫
身鳥卑軀舫舫舫舫舫
反射艇舫舫舫舫舫舫
向返舫舫舫舫舫舫舫
自邀履舫舫舫舫舫舫
所血肩舫舫舫舫舫舫
的邊舫舫舫舫舫舫

尸部:

尿熨改彊
屈舉張已
壁辟強異
刷咫選弼
屍劈貴弛
慰犀已弧
遲屎丑弼
尾屈眉弓
尺遐巴弗
屏尉槩弼
避腎慾弼
尼屈屏彊
屈屈屏彊
層層屏彊
屋層屏彊
居層屏彊
屬層屏彊
展層屏彊
局層屏彊
尸層屏彊
民層屏彊
履層屏彊
尸層屏彊
引層屏彊
鵠層屏彊

衤部:

裕祺裨搭
族裙褶旖
禮禱褶旖
旅裸各祗
初禪袒祗
裡祿祁爛
神衫祀祗
補襯袂襠
遊祈素袂
福禱袂袂
施旋祐卞
放袖揭謁
視禍祠謁
社祗襟謁
被復袂謁
產祖袂謁
文祗襟謁
示祗襟謁
方旗襟謁
於祥袍襟謁
衤襟謁
衤襟謁
衤襟謁
衤襟謁
衤襟謁
衤襟謁
衤襟謁
衤襟謁
衤襟謁
衤襟謁

宀部:

票害束宮酸牢宰醋迺轆穹軛軛
官輕露震宋曹窠輒輒輒輒輒
專宜宗宏窗甫惑窺窺窺窺窺
更究麗窺窺窺窺窺窺窺窺窺
平擊守雷宵寓寡八寥酣窺窺窺
或速零宅寮窺窺窺窺窺窺窺
頭察歌賓醒宇霍醇酪賓寐有體
再容融哥豆霧賈軒寢寢寢寢寢
安額突寫寒割飄軛由潛輓輓輓
實需密輔寄宴窩醞輓輓輓輓輓
東配寶暫栗醉寂酷棘齡酃酃酃
案轉富雪寧逼霞斬醉粟罪輕輓
兩完副賴覆輓酌甄寅吏邇醞醞
車較雲輪塞輓輓輓輓輓輓輓輓
電客宜寬刺輓輓輓輓輓輓輓輓
定賽室輪軟丙軌窄冥賈窪帘剌
事整審憲輓輓輓輓輓輓輓輓輓
家連字輓輓輓輓輓輓輓輓輓輓
要空爾它穿繫敷宛霹靂窖救輓
可西遭惠遷轄魑魑魑魑魑魑魑

害羣鄺醜通勇愚甬勁頸逕亞惡翌雅牙邪鷺鴉 279

禾部:

和乘稻穰透 我秋吞秧穆 利稿稚迂 82 千秘憩稗 種刊乖馥 委稽黏稜 程稍穆壬 科黎梨孟 稅剩愁孫 香丟鵝秣 稱廷穫黍 移魏穗舐 積秩禾秣 升秒頽稷 透于稔穉 私喬禿忝 季舌稠稞 租甜天秬 秀稀稼稈 穩刮犁穉

犬部:

獲狐 犯狡 獨猿 獄狸 獅猯 狗獺 猛猾 狂狩 猶狙 猜獐 獵獾 狄狎 狹狽 狼狽 猴遯 狼狽 猩獐 逛獠 猖獗 獾 39

門部:

開刁承習翟 司闕孩群帶 間闕孫盡聿 關闕疏尋弄 問闕孔既良 門闕孟逮 85 閒闕孤君 閒闕遜函 關闕攷肅 閉闕丞羽 闕門孺翠 閱闕子暨 閱闕子暨 闕闕建亟 閃闕即尹 闕闕畫郡 闕闕書翌 悶闕那戮 悶闕了退畫 闕子劃戲

八部:

會念鎖貧俞較鑣鉛鑄錫銀公 分錄鏡鑽緩鈣錐銖鑄錫鵠企 八創鈞倉鎔銀鎔鑄鉤創鑄翁 金鐵鍾鈔鉀鎔鉅鉅鉅鉅鉅鉅 合拿逾劍鎔鎔鎔鎔鎔鎔鎔鎔 全針舍飼鉀鎔鎔鎔鎔鎔鎔鎔 今錯欽銳鎔鎔鎔鎔鎔鎔鎔鎔 入愈盒盆鋪鎔鎔鎔鎔鎔鎔鎔 希飯~既餓忿鎔鎔鎔鎔鎔鎔鎔 餘鐘頤貪飢鎔鎔鎔鎔鎔鎔鎔 銀途鍵鍊鎔鎔鎔鎔鎔鎔鎔鎔 鎮鋼銅鈞禽鎔鎔鎔鎔鎔鎔鎔 銷飾敘鑲鎔鎔鎔鎔鎔鎔鎔鎔 令飲餅鋁鋁鎔鎔鎔鎔鎔鎔鎔 命錦鉅斜利鑲鎔鎔鎔鎔鎔鎔 領乏鍋釘鎔鎔鎔鎔鎔鎔鎔鎔 食鑑飽鑲鎔鎔鎔鎔鎔鎔鎔鎔 闕闕畫郡 闕闕書翌 悶闕那戮 悶闕了退畫 闕子劃戲 殺銘舒鉛鉅鉅鉅鉅鉅鉅鉅鉅 館余錫銓飭鉅鉅鉅鉅鉅鉅鉅 錢含鈴蝕鑲鉅鉅鉅鉅鉅鉅鉅 錢鑑飽鑲鎔鎔鎔鎔鎔鎔鎔鎔 命錦鉅斜利鑲鎔鎔鎔鎔鎔鎔 領乏鍋釘鎔鎔鎔鎔鎔鎔鎔鎔 食鑑飽鑲鎔鎔鎔鎔鎔鎔鎔鎔 闕闕畫郡 闕闕書翌 悶闕那戮 悶闕了退畫 闕子劃戲 239

火部:

業步忙煩爛烘灼样惋繁檜松莖 對止炸煌悅驚敵敵愾愾愾愾 也榮慢怡恨雀恃烽爐愾愾愾 此勞恢恆蕞斃熔道愾愾愾愾 小快煙兌紫螢燬炬燬燬燬燬 當順齡憶卜憐焰憐憐憐憐憐 情尙怪懂愉惶怯炊慄慄慄慄 水州炮煉煥燥忱悸憎慄慄慄 性燈川齒恰燭雌悍燬燬燬燬 少幣燃憤燬燬燬燬燬燬燬 收堂慎憾惱鑿燬燬燬燬燬 營懷憤歧佈炫懈慄慄慄慄慄 黨掌爐柴肖焜焜慄慄慄慄慄 常輝炎嘗煤餉恍燬燬燬燬燬 省賞惜削慌愧燬燬燬燬燬 些怕耀炳慨悼灶悃悃悃悃悃 光爆尖弊悔悃悃悃悃悃悃悃 曾頻惟懼悃悃悃悃悃悃悃悃 歲燒慘烤悃悃悃悃悃悃悃悃 火肯炒劣悃悃悃悃悃悃悃悃 251

女部:

如戈划廷嬌奸 好姑媛弩嬌姘 女怒姬嫵嬌姘 她娟姨嫵嬌姘 嫌姻娥嫉嬌姘 始娛媚嫵嬌姘 飛姆媚嫵嬌姘 姓姊奴弋姣母 婦婉盡妒嬌母 婚奶妓媽嬌母 媽媳怨娣嬌 努娃奸妍娣 妨妮妃姥嬌 媒婷嫂婢嬌 姐妖娼妊嬌 妹嬌娼娼嬌 娜姦絮娠嬌 妙嫁帑嫵嬌 妳姚嫵嬌 娘嫩姐嬌 210

艸部:

落勤蒙荒鞭苓蒞葵萃繭纒在服博
若散勤芝擎蒜欵靴蓄蔥鮮纒首協
蘭苗茂邁菁蘿蘊旬鞍孫軟變鶴繡
英莊荷欺蒼蔗苔菱萱蕊觀鞘藥薑
世運藍萍委肥蘋正莖葦蒿莧薺薊
帶勒夢蔣蕩蓄蔚茵次辜莆茴覃薺
難苦菲薄蘆蕪葵芋倍葆苟巨鉅繁
斯舊茲萊蕉薛荀萌孽鵲菅纒薺薺
廿菜蓋莉蒸荀萃蕪芥秀莢莖薺薺
花某巷募苑昔芒署藐荔蕪薺薺
黃葉莫燕薺蕪蔭暑莞莘鄧苒苒
著重茶范薺蓉茅藻芹茸蓆苟苒苒
觀蔡蕭芬蔬菊苟苒苒苒苒苒苒
基草革葛戲蓬茄苒苒苒苒苒苒
共藥驚菸陸恭蔥采芸葫蕪高絲半
華歡藏勤蒂芭茫微菩鞠薺孤薺
期甚藉敬墓沙芽薑苞蕃藕荻蕪
萬藝新鞋園藤莓蔽葵蕪薺薺薺
警卅幕甘蒐艱蔓斟蒲芥菜翁莖苒
其蘇芳艾蓄葱藩存荆蔑藹波菽秧

280

一部:

馬歷巨奈辱辰奢戊贅番蜃硃殃耶
感盛鬆奔雁碳驕磊礪甬夸敵殞聘
否緊奪騙礪爽矽魔礪硯仄狙殉耳
研右麥灰礪碌硃馴驥神壓臚殖聖
太尤丁碰臣吾鬆盜愿砧駁尤裂聚
醫左厚夾驟碰硃礪礪硃鬆駁殊聽
雄臨賢礎臥腎歪厄磴戛斫匏歹職
原監豬砲碑硃礪礪駁駁駁駁駁
面歐硬奮馳唇豎礪駁駁駁駁駁
區驗夏丈驅郁匠鑒硃駁駁駁駁駁
百死髮礪礪駁駁駁駁駁駁駁駁
五存勵礪駁駁駁駁駁駁駁駁駁
而願駛砂駁駁駁駁駁駁駁駁駁
來壓碼耐礪奎歪丕臂駁駁駁駁駁
成破駐騎犬匹熙駁駁駁駁駁駁駁
長確套憂碩孟威匪駁駁駁駁駁駁
大石堅鹽匪肆硃駁駁駁駁駁駁駁
有友奇曆厲欣厝煩減戌駁駁駁駁
在威匯礙碎匪威駁駁駁駁駁駁駁
不布逐覽磁厭巫遠匡匪疋查瓠殆聰

293

才部:

批措攏搖寸披扒拭捺擣捫
抗振頂揭描抹扶捺捺捺捺捺
切招捷哲抑挾挾拮拮拮拮
括折援擠拮拮拮拮拮拮拮拮
擔擬拓挫挫挫挫挫挫挫挫挫
排攤掉撥擦拮拮拮拮拮拮拮
技揮撤掩撫挽扎拌扔捺擣
採捐握抓攏攏攏攏攏攏攏攏
把搶援拔搞撤擣擣擣擣擣擣
推控操扮撲攏攏攏攏攏攏攏
打拍探撞捲捉拌抖拮拮拮拮
據攝攏搜攏攏攏攏攏攏攏攏
求播拒挑抬捨揆揆揆揆揆揆
接揚擇掃拮拮拮拮拮拮拮拮
持捕抽掛摸攏攏攏攏攏攏攏
拉攏授搭拮拮拮拮拮拮拮拮
投損按撥托扭逝攏攏攏攏攏
提救扣拖擣拮拮拮拮拮拮拮
指找拆抱擣拮拮拮拮拮拮拮
爲換抵押扶拾擣擣擣擣擣

231

入部:

育庭廖帝麟廂瘤孰變患
率座旁削疲庚辛庖痠瘰
廠慶雜辛廊玄瘰瘰瘰瘰
度武亡唐廈瘦齋竣瘟疔
辦重序摩廚韻妄敵痰痞
交夜棄毫彥盲靖壘韶痢
商站廢床廬廬瘰瘰瘰瘰
就適竟顏瘰瘰瘰瘰瘰瘰
應亦庫鹿襲廁頑疵瘍瘰
意離郭敵率亨竭瘰瘰瘰
府音刻廟磨殺遮瘰瘰瘰
重病章端瘰瘰瘰瘰瘰瘰
立龍衣熱疾疫裏云亥瘰
六席療亮氓痕辣瘰瘰瘰
動親親敦瘰瘰瘰瘰瘰瘰
新底京齊瘰瘰瘰瘰瘰瘰
元店彰症豪郊瘰瘰瘰瘰
部麼康痛瘰瘰瘰瘰瘰瘰
高効廳麻忘動瘰瘰瘰瘰
市廣充享衰敵瘰瘰瘰瘰

31

琳琨環闥 瓊邸號門 瑜琢砧闌 玻璃瑣闥 弄瑣環環 碧琉璃璫 琴瑕剛璫 玲珮瑤瑣 珍瑣珀瑣 珠瑣璫珪 珊瑚璫璫 阮琦琶那 瑞琛琵琶 玉瑣璫瑣 班瑟玷玳 85 環玫瑰瑣 球班胡瑣 王琪瑾玖 瑤瑪琅瑣 瑤瑣漢江式

隔隘 阻併 陣隅 陽陡 障阪 階陞 險墮 降陌 附陀 隨陋 隆隙 阿阮 限墜 防隧 縈陵 陸陶 除陰 隊陪 陳隱 荒陷

加力架刀駕忍賀刀迦袞又桑又蚤發登鄧堯癸心
必 21

備傷盡袋仰俠兜仵備有條
 何值倒傑傳儒侶爪傲僭德邇
 伊你伯隻佩爾情俯伏倔排駢
 信係什併偽療伙儉伽信仝卯
 件修候佛伏仗岱伴萬丁及卯
 仍例促仔焦叨戴倖侈齊僑迎
 任舊延侵伴伐傲倦抗倭情貿
 代便偵偷仙僵俏僕佃僮俵留
 保停仁橋瓜爬侍佣什倭尙劉
 價份似偏俱仇倘儂饒哩仕僇
 使貨付俊伍戩伶傭僑俸僉僇
 化億佳儀伸催佑倖伯兆空僇
 們住健俗仲仿俾攸傀僇有僇
 立低估借偶佐貢僧傢佗儻僇
 進傳像償側侯倡奇僇失僇個
 但條休貨偉堡悠辟儼儼倣哀
 個集假佈諸僱俟倪每倣僉僇
 作依優價愆孚仕同僇仆僇佳
 他做僅倫傾遞傍優倪佻俺閒

勢嘉幸埔塹墳翹級塹擊乾塹未幽
老刑趨趙埃穀堯督圻邁韓李祀容
增青赴址貢趙壹頡駝塊裁載稱齡
規超趣彭坑塢卦垠緒峙幹羣毬鹹
士熱壞劫寺埤壩邀塹劫胡齡耦卓
項功鼓塊壘盤墩烹垣索韌遷貞
達境趕末赤溫脊珂毫垓朝烈丰桌
教夫圾坡堆坊臺垣坵坏古兢耘占
去賈垃壽趁袁塘罄蛭玃故哉稔餐
起執毒替壇煮圻敖吞圩違臻秉肄
未聲吉敢坤馨塢堰馨聳真幹垂匕
都型恐巧奉卉耆塚穀墳致翅耕顛
場責越干堪圳垢塹塹聲直韋耗旨
報春志坦塔秦豪塾埂聲支翰郵穎
表考喜培孝堵圭邢赴壘克辜拜頃
工土攻封墳般垮墟燾垠南勃邦皆
天均素域井赫坎壺貢圩至裁毛疑
三款泰郝坪墊熬汞釐警到戴看比
者走城幫奏臺赦懿圻墀二截手七
地形遠靜埋堤壤贅蠹遠一載餘稻塹

衡微徐衝徒律徵街衛待復須術很德往從得後行
衡微徐衝徒律徵街衛待復須術很德往從得後行
衡微徐衝徒律徵街衛待復須術很德往從得後行

崩濤嶺 岩嶽峯 凶嵐峴 崔顥崙 嶼巖崢 豈嶠嶸 幽峭剗 崗崇岫 崱坎嶮 嶺爰峪 兇嵌峒 峽峙岷 崇峨嵕 凱峻岐 乙艷嶺 峰嶄巍 岸炭崧 豐崖嶠 山崑岑 出崎屹

66

目部:

則助置遇昌旺鼎賊膨髓囚幟豚點喧飢腴輻胃收
 股且署腳帳罷嬰賦肅肺旱瞬健膾肛鸚舛腴賁
 點具盟略貼賺賜穎畏腺畸賬貽骸曦腑圍胴鳩肫
 昨財勝朋骨睡膽晶罕巾瞧賤圍貶臍憤胝馬暮均
 明服周野冠夥胸豔昂匙睜晰皿晒膾點朦瞶膈
 最團申肉剛默壘昆脹曝肪剔國臆幌敵瞠幡脾
 目界週甲賄冒腹罵晴晃曬瞞晏膝惟响邊睢睪
 同農圖販畢貽騰腸罹遏怕聖胚瞭睽臨涼胯腓
 縣眾星困膠肥暢臟冤罩幢翠帖賑腰路吳胛青
 內見照思固圈迴脈腔疊曠冥晃眇胭路疊眈曷
 過量早晨罰恩曉腰曠貶脾膀罔脹曳膊苦朧
 月題晚異贈暗離墨胖瞻畔味瞪膳晦煦听臍
 用還景田映胞冊邏胃瑛賦瞄盯冉眶脯潰晚
 四影購圖典曼胎脂肝腫膝瞎辟曰眩矜眸影
 由回買羅旦臉昭鵬睛鶴愚暉曜骼肋噎形瞻
 因運顯幅敗累腿膜贖瞎暇腕迴黯除胰替骷
 日果里罪脫丹迪罔脆麟臘貯昱曙睫蹂動瞋
 時數男黑圓貝鴨刪肌肢睦腥羈用囿蜀臉肱
 國體園腦曲昇盼懸帽歇帆肘毗嚙般腋腫體
 是軍易浪暴療媛滅壯氓暑球量腔裏暖暈暈
 沙掩

口部:

味跡吹哪哄喋唆邑嚙叨吩囁唧吒嗟蛔噫
 吃噸跨咬吟喇咕蹊噴喫爾驚蠕躡蟲姪姪
 吸嗎噴啊啞蚊喃蟬鳴唉嗽嘍嚙啞嚙啞
 央呢嘴啦吻鷺叮哦咒蹦蹂吟啄踴蝗吁炸
 足嚇吊噎逞蜩蟻貽咪咚嘶蟋踴挺嘹蹇
 嚴距踏凸虹蚶躁嗅咱嘿娥蝠噤蛹螭燒
 器唯啡蛇蝶蟻蹻蛙吱噁喳吠嘍吶錫
 吳跑螺呖喻喧盜哼噲呱咄蟀跛蹻戕
 雖跟喊吵踢凹囉咀咋躡踞噤噤蹻蹻
 另跳吧噪啤噤噤噤噤噤噤噤噤噤噤
 別遺鳴蹻蹻蹻蹻蹻蹻蹻蹻蹻蹻蹻蹻
 號呈蟲喚喘啞啞啞啞啞啞啞啞啞啞
 口患蜂嘆唸訕訕訕訕訕訕訕訕訕訕
 只忠串蹤哩嚙嚙嚙嚙嚙嚙嚙嚙嚙嚙
 單貫躍咖默蠟嘲哇嚙嚙嚙嚙嚙嚙嚙
 戰叫哈蹟呀蹂嘔嘔嘔嘔嘔嘔嘔嘔嘔
 品呼呂哭踐跪叭嘯嘩呐哺啞啞啞啞
 路唱遣吐喉哈喀噓嗽顎媽啞啞啞啞
 員跌兄蹈吋嗜噎蹄叩噓傍叭啞啞啞
 中史喝蝦呆哨鵲吼助蚌蛻叱嗤虫蚶
 孟蚶蛻蚶

丿部:

采鄱冬 貓貉狗 貌狸久 乳爰然 屯谿各 75
 妥百名雒 辭邈外黃 番豺多殮 雞刈認鷄 悉采蠶架 翻柚彝舛 釋奚彙駕 頓貂煞灸 亂孚剝炙 平約詹咎 彩舜脅遙 愛爵危祭 爭想急夕 受覓色怨 人塾貊逢

ㄥ部:

繼絡縹緗紅 紀純紋縹緗 絕緩幻縹緗 斷絲縹緗 紅縹緗 維紐縹緗 系繳縹緗 幾縮縹緗 響綜縹緗 級締縹緗 續紙縹緗 給細縹緗 線編縹緗 約網縹緗 結織縹緗 鄉納縹緗 統練縹緗 組縹緗 總紛縹緗 經終縹緗

米部:

讠(言)部:

彡部:

风部：

木部：

34

勹部:

包 旬 忽 欠 旬 勿 勾 勿 鄒 旬 勻 黠 黠 旬 勻 創 芻 勸 匍 匐
刎 鋸 22

说明书附图

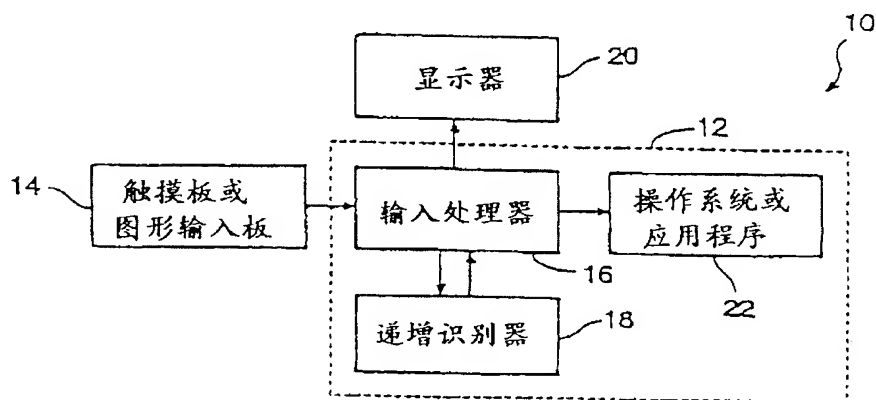


图 1

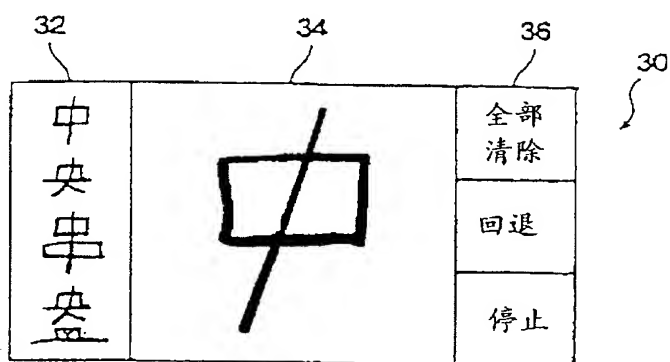


图 2

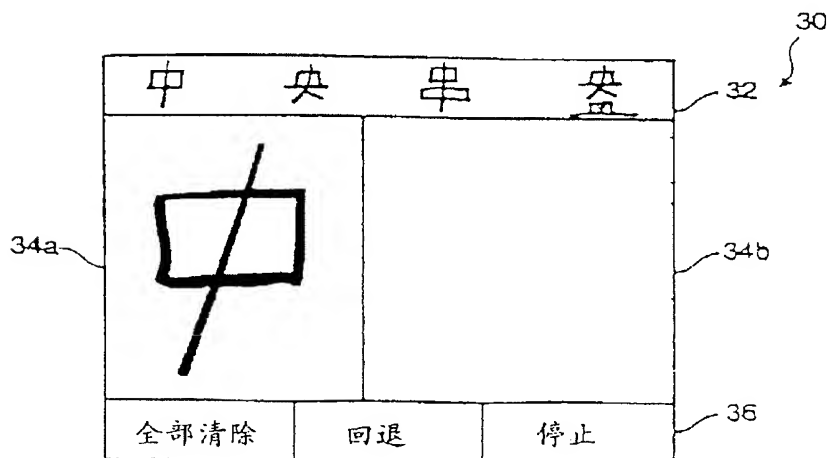
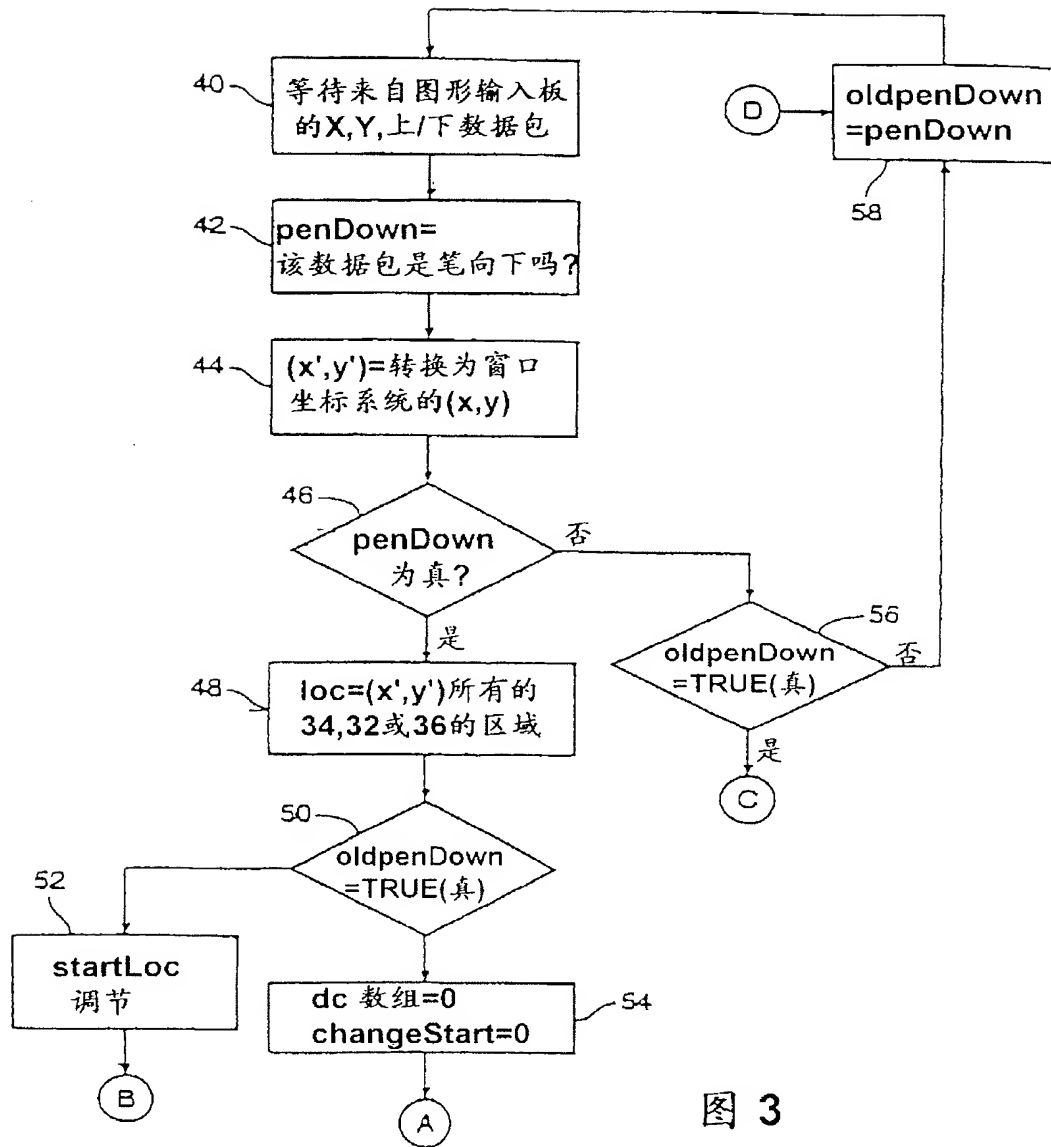


图 7



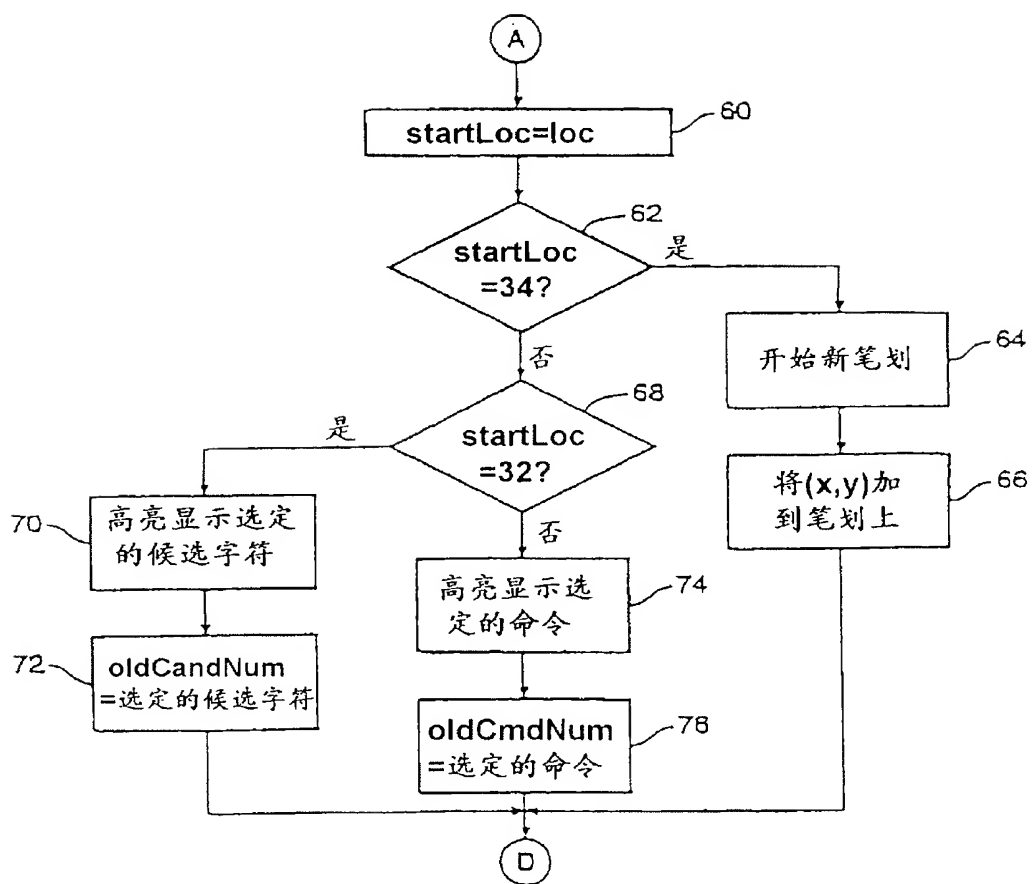


图 4

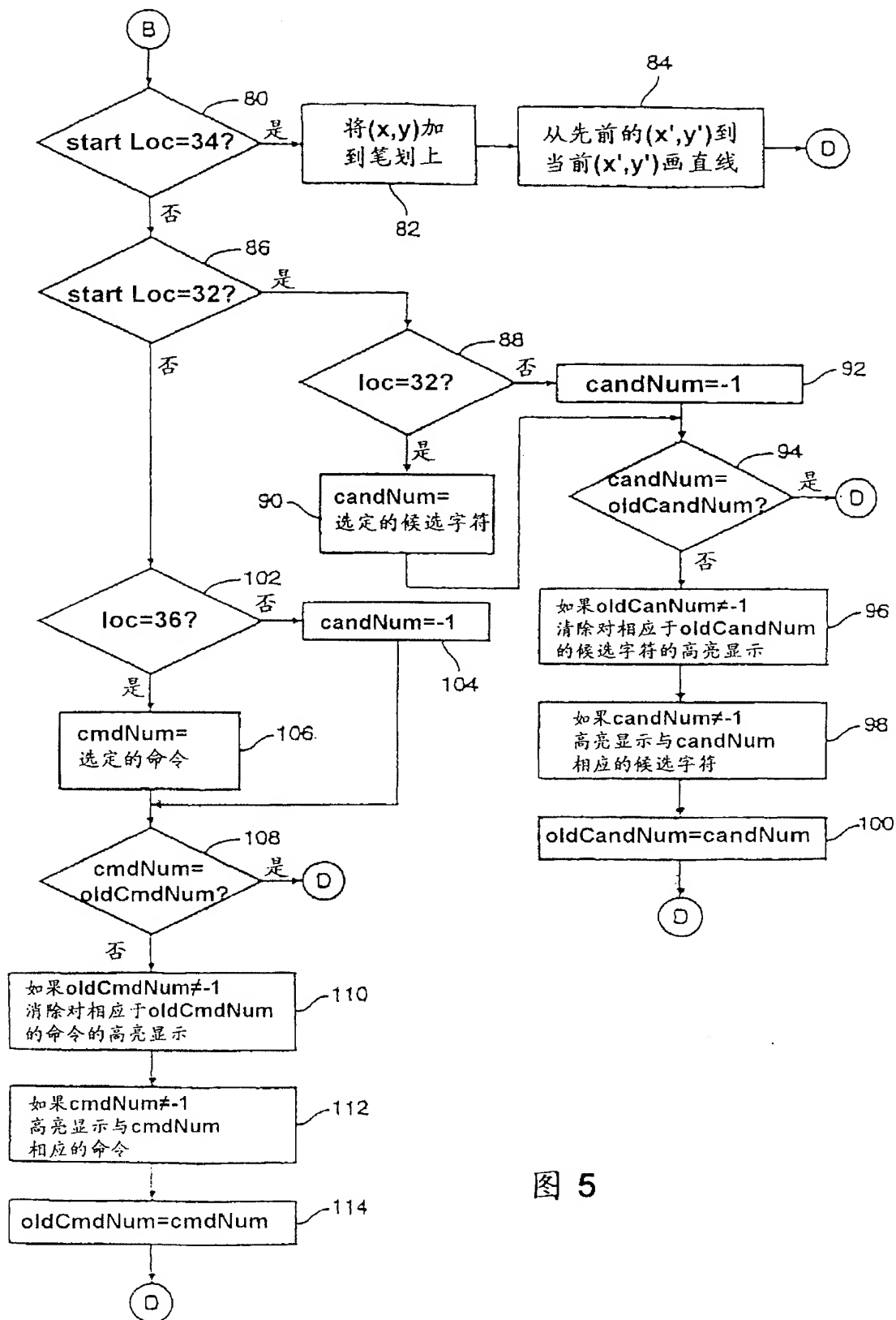


图 5

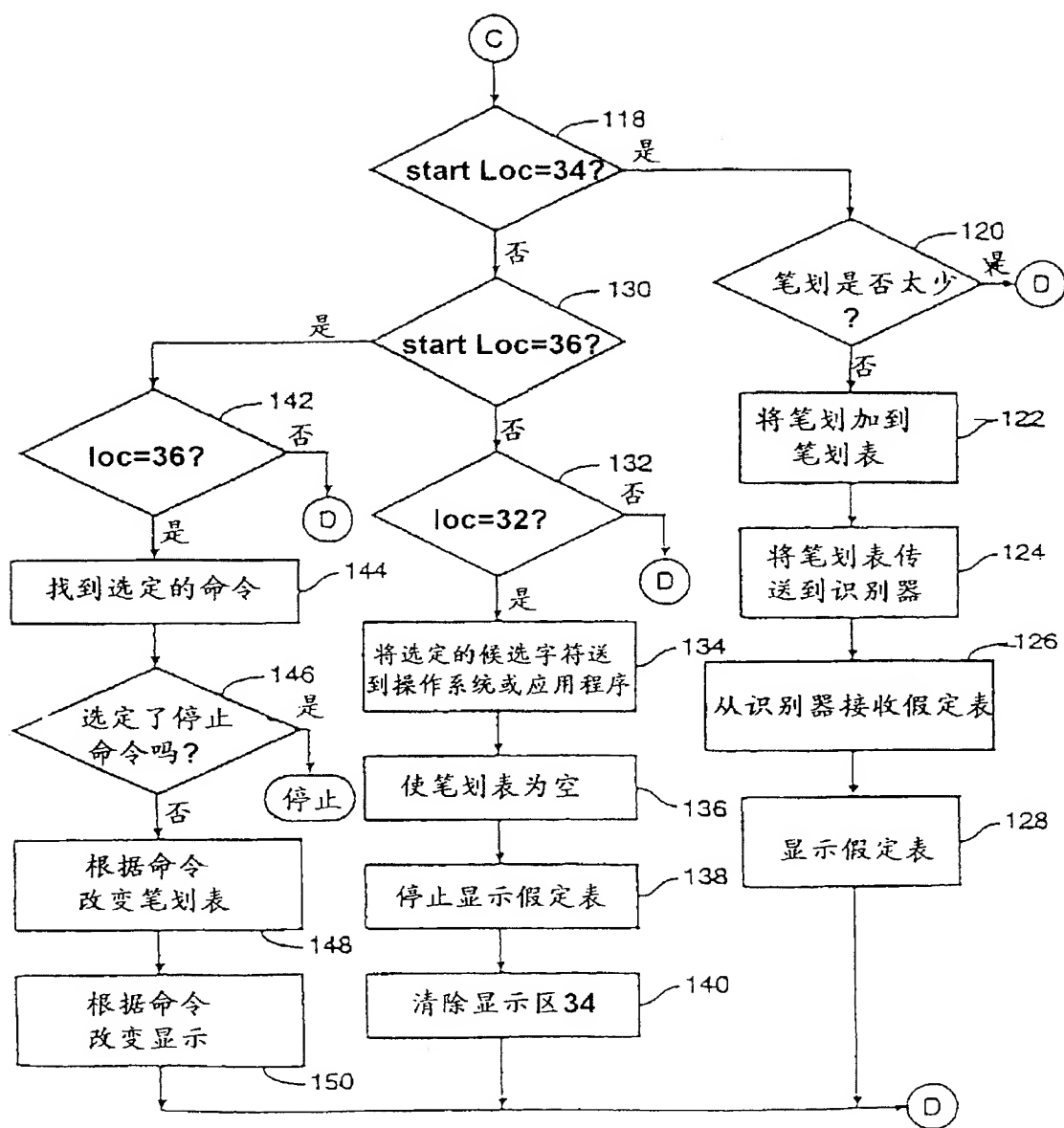


图6

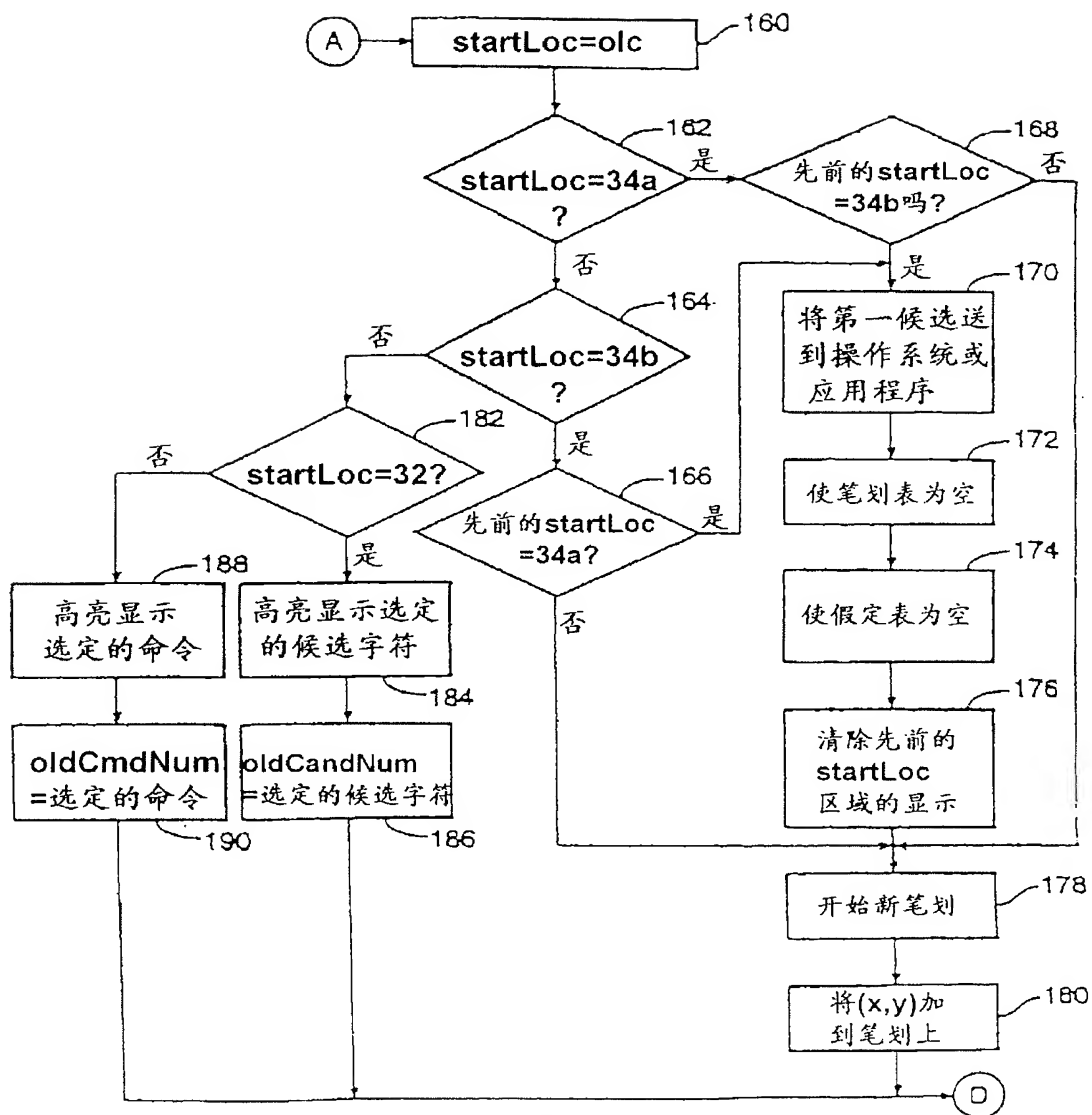


图 8

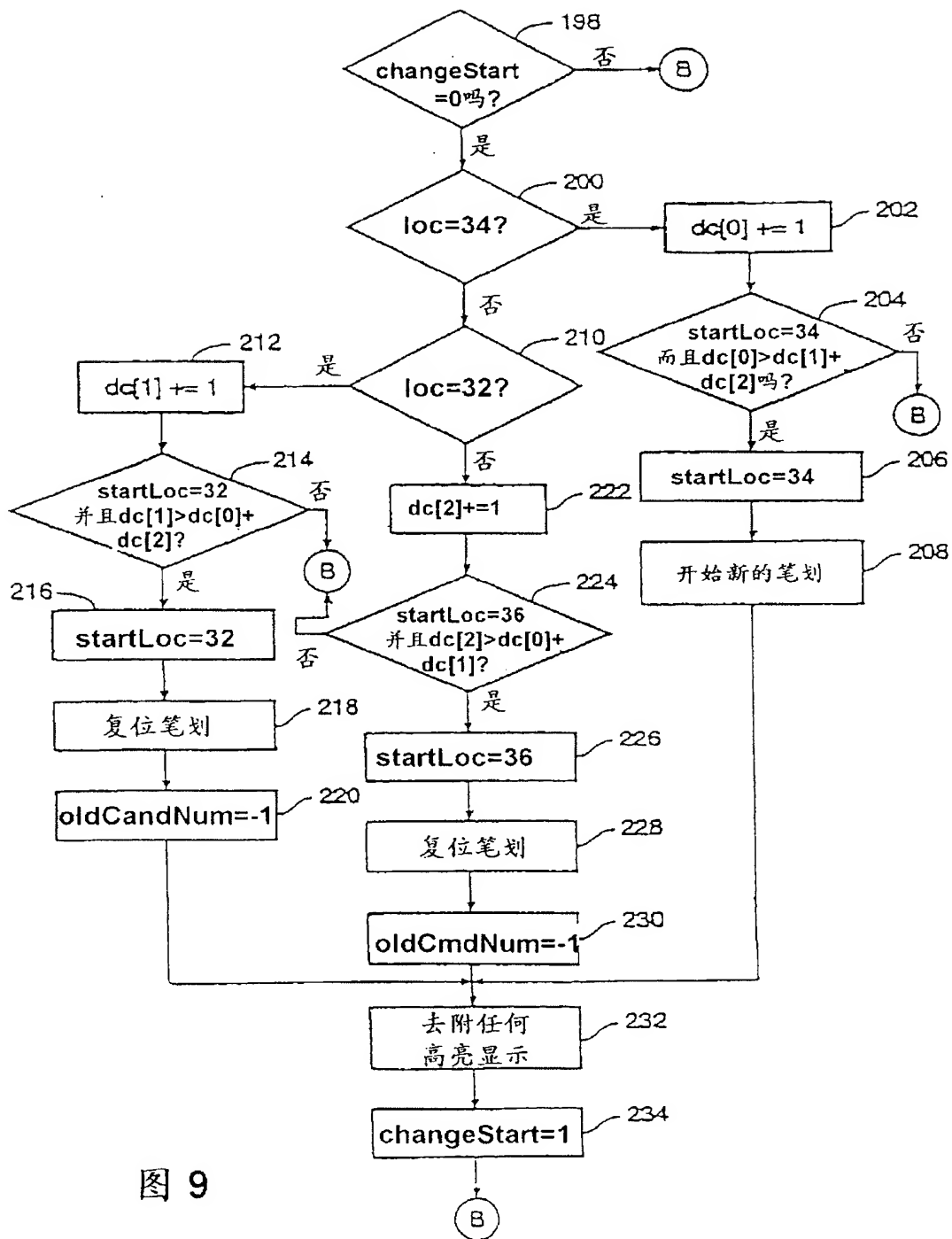


图 9

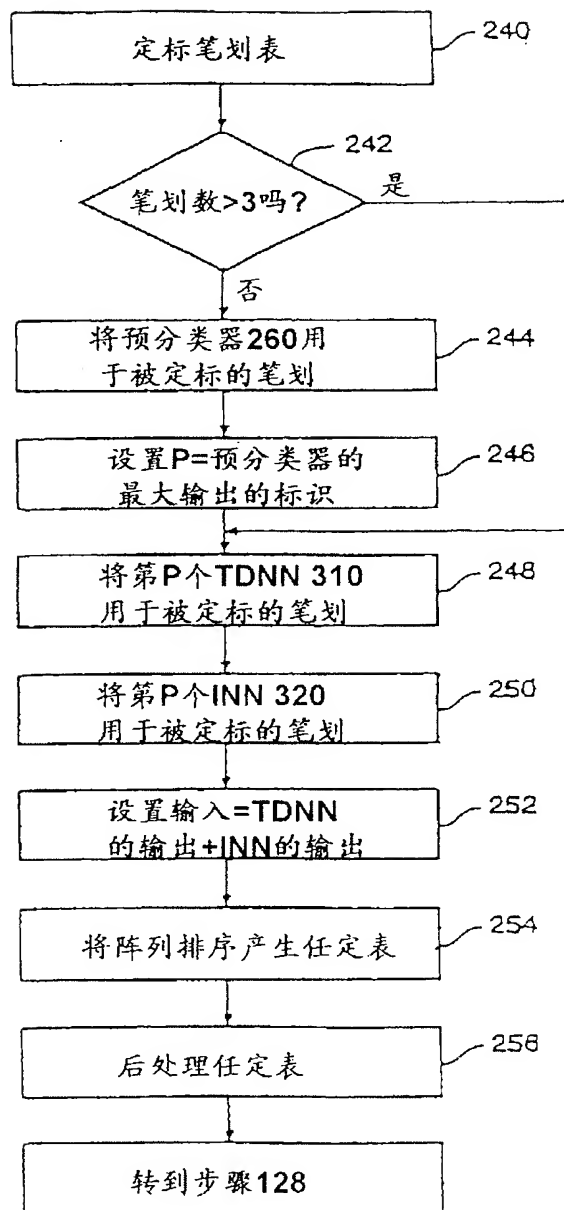


图 10

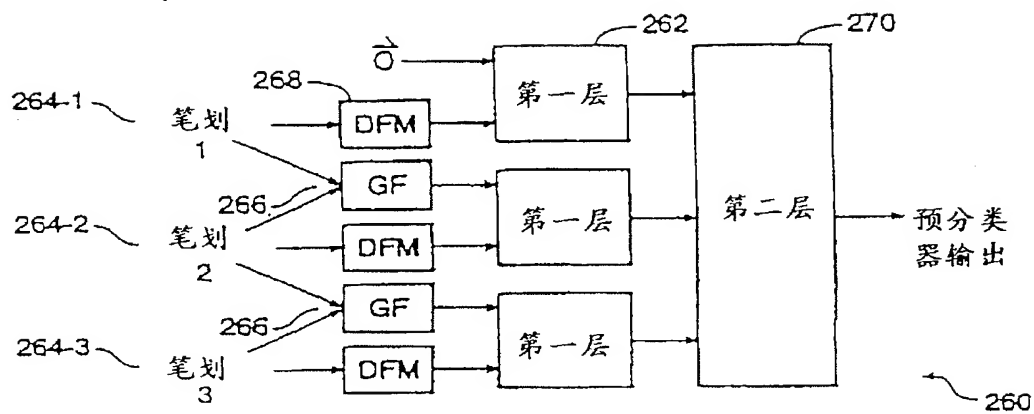


图 11

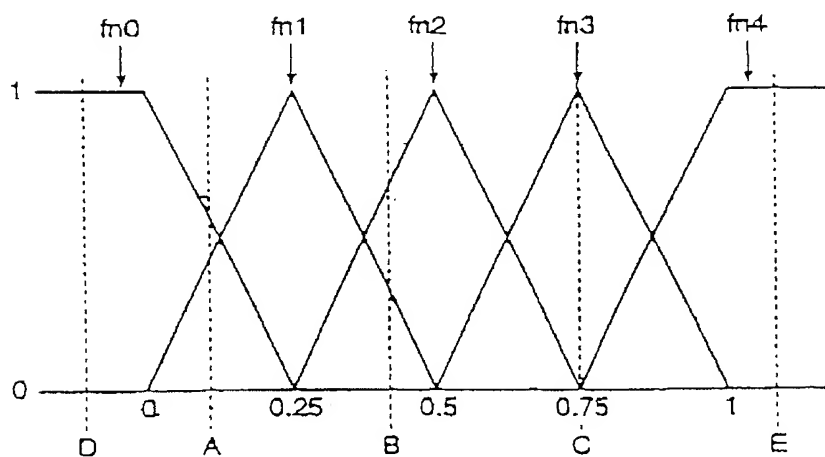


图 12 A

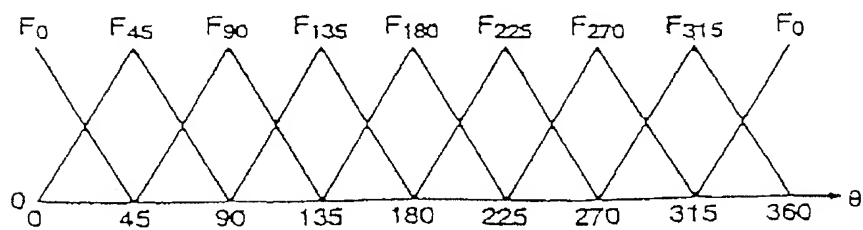


图 12 B

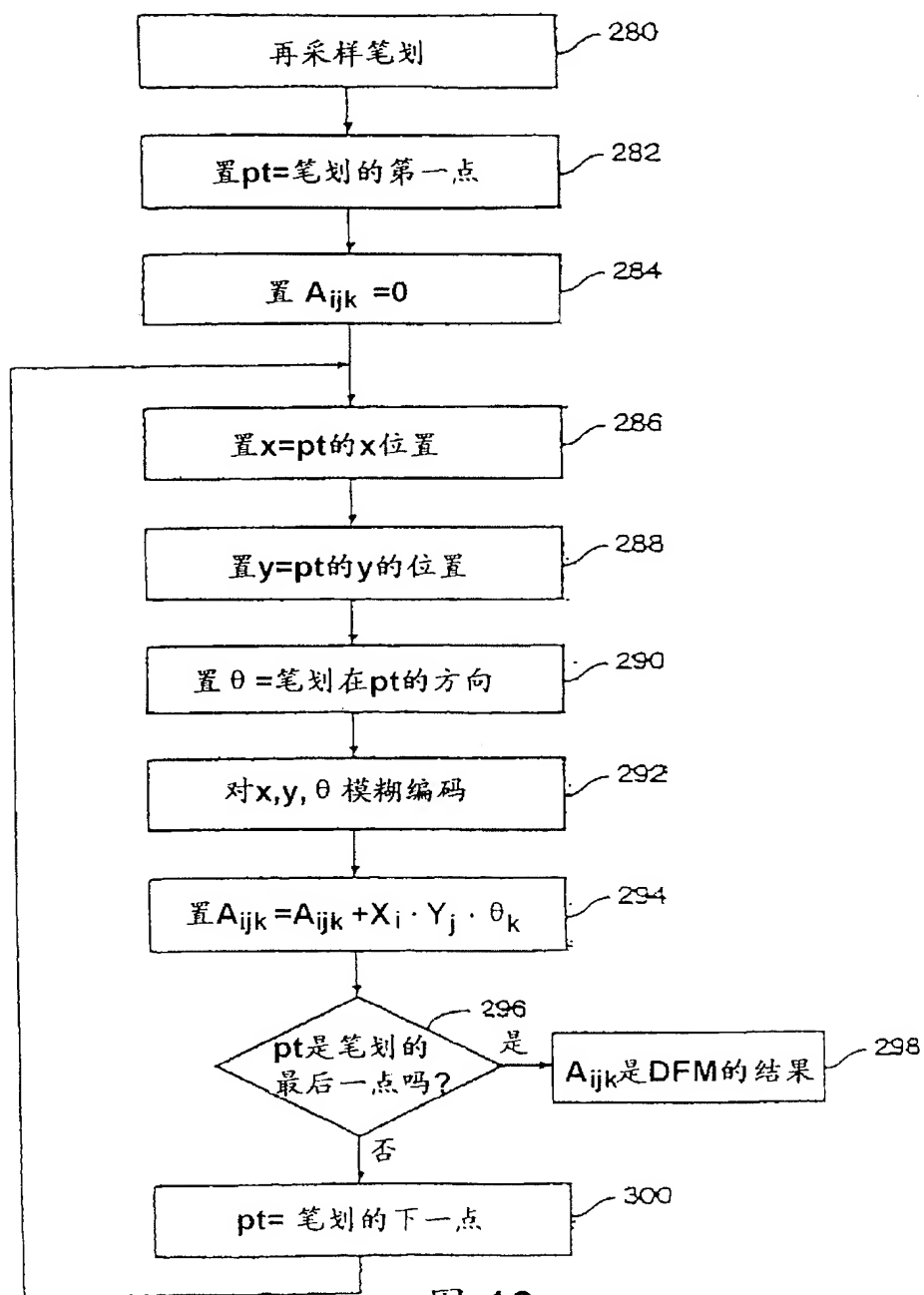


图 13

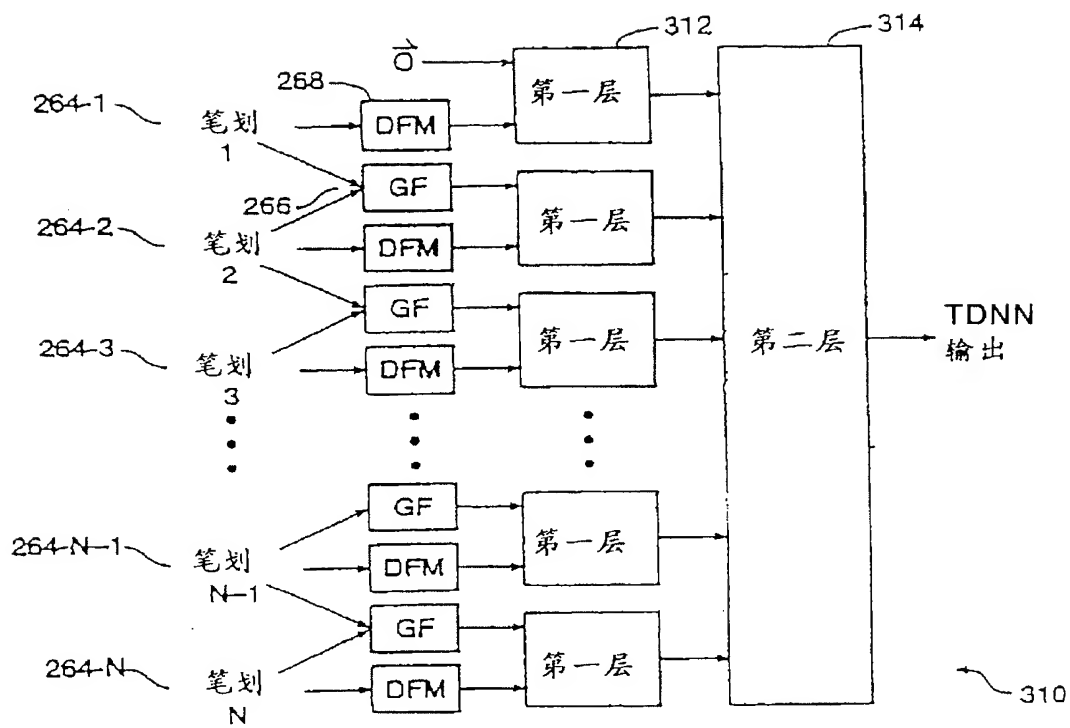


图14

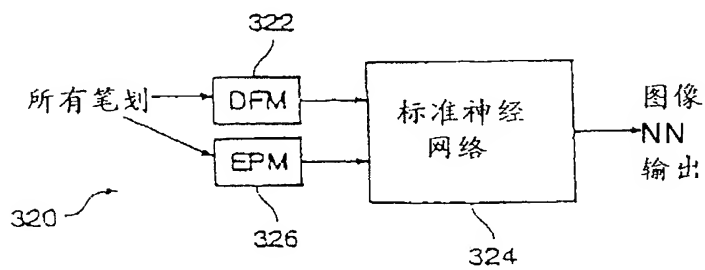


图 15

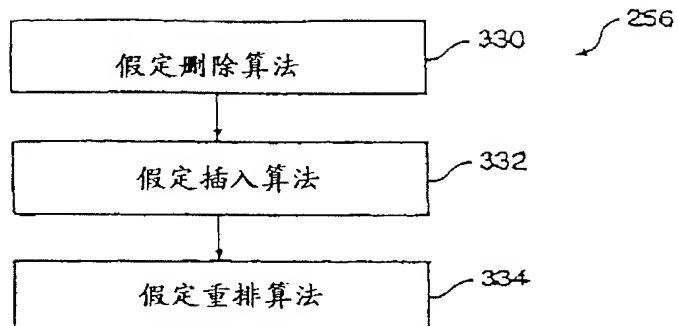


图 16

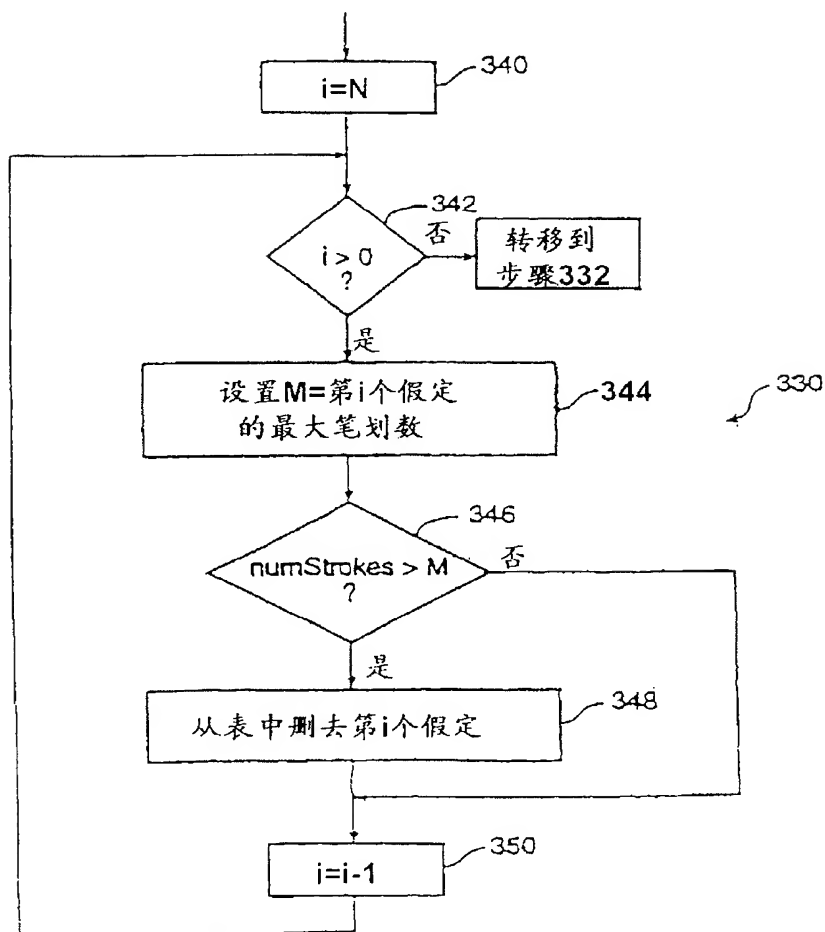


图 17

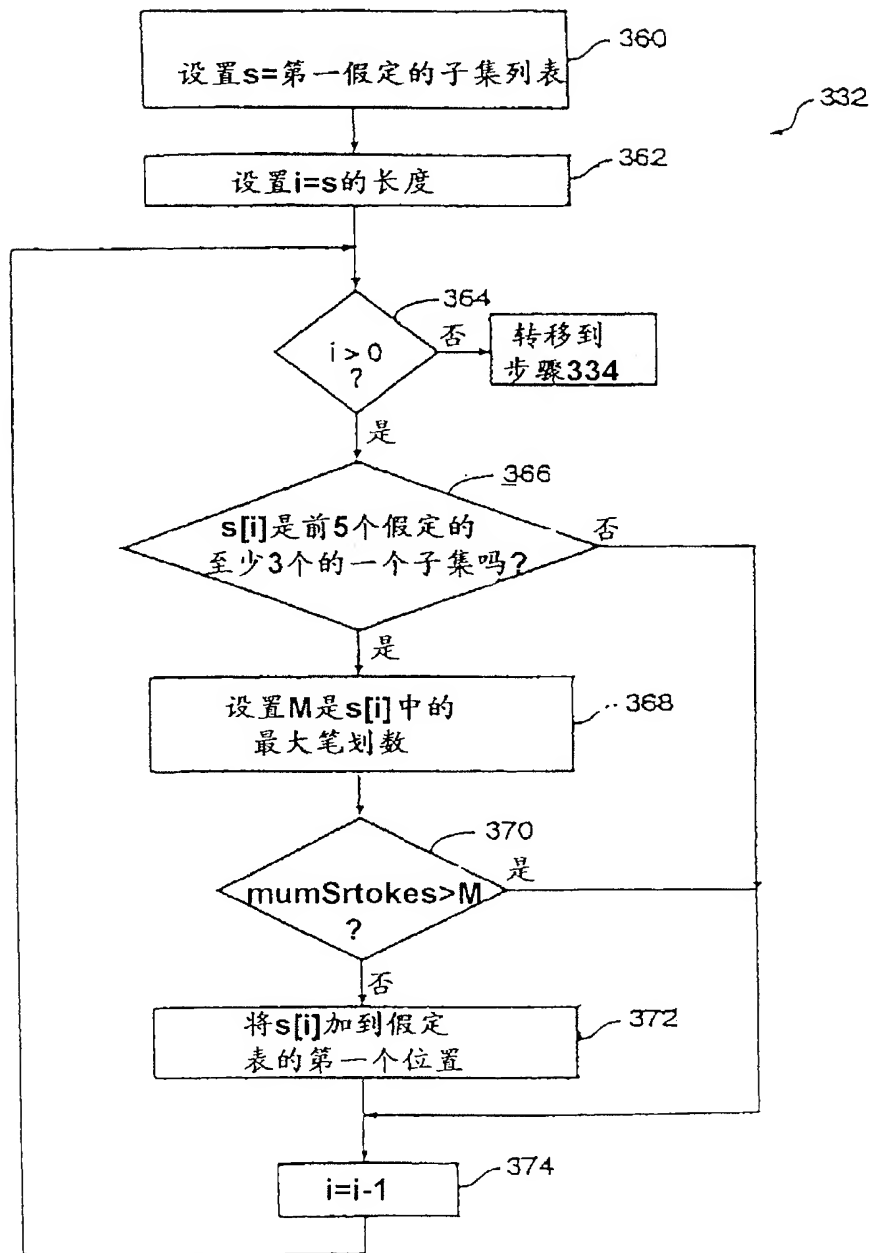


图18

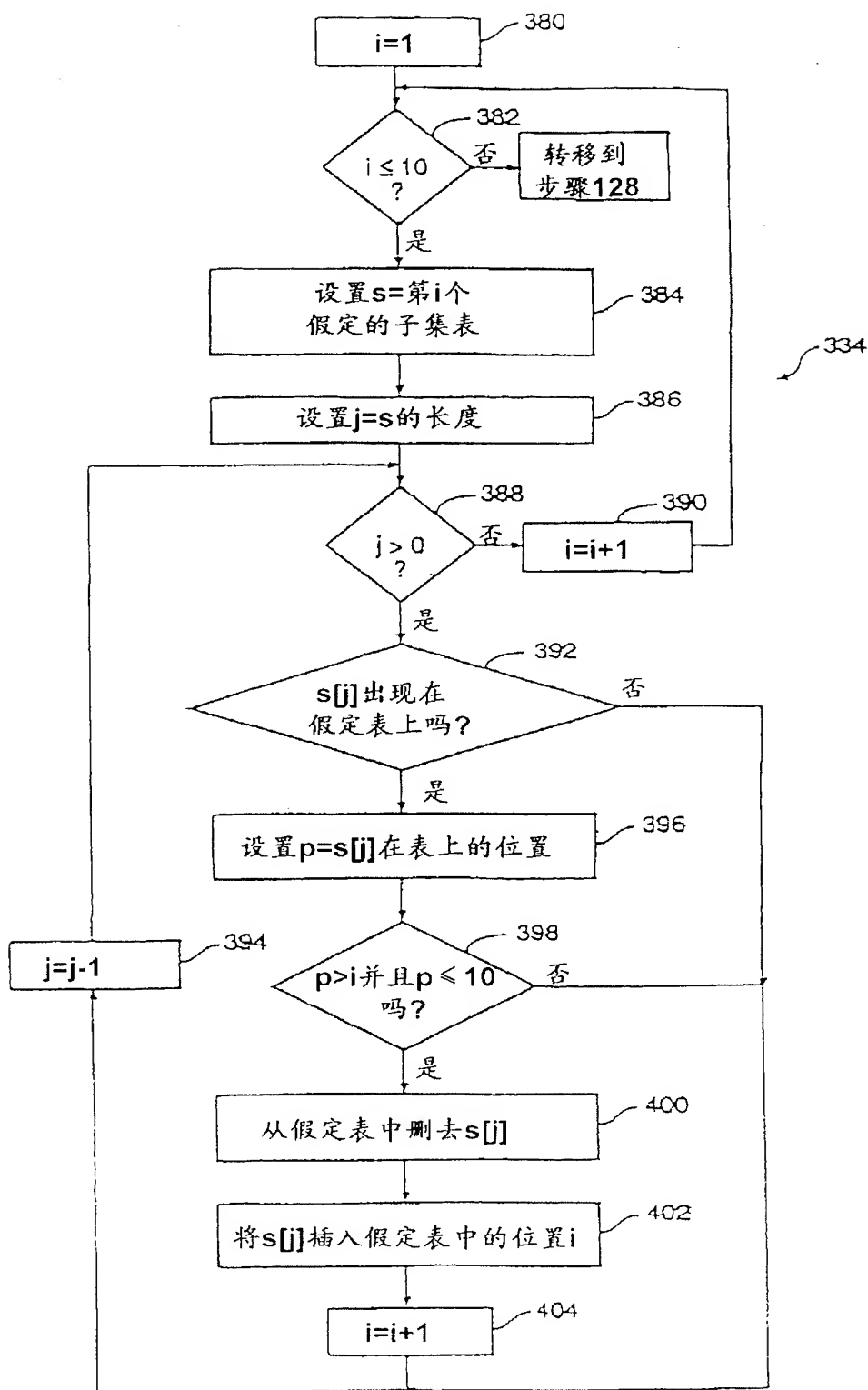


图 19